

INVESTIGATION OF THE INTERACTIONS THAT GUIDE
VASCULOGENESIS AND ANGIOGENESIS IN THREE DIMENSIONAL ENVIRONMENTS

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

In Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Anthony Diaz-Santana

May 2014

© 2014 Anthony Diaz-Santana

INVESTIGATION OF THE INTERACTIONS THAT GUIDE VASCULOGENESIS AND ANGIOGENESIS IN THREE DIMENSIONAL ENVIRONMENTS

Anthony Diaz-Santana, Ph.D.

Cornell University, 2014

Tissues require that their cellular components interact to gain functional structure and to survive. Fully dissecting the myriad signals experienced by individual cells in the microenvironment *in vivo* is an ongoing research challenge. Elucidating the guiding mechanisms for cell communication may enable the rational design and manipulation of living systems on the tissue scale. In the context of tissue engineering, particular interest is allocated to the control of the cell-microenvironment interactions to build healthy thick implantable tissues. During vasculogenesis – the self-organization of vascular networks – individual cells undergo morphogenesis and assemble interconnected networks of capillaries formed of endothelial cells (EC). In addition, this primitive network sprouts new vessels and further remodels (angiogenesis) resulting in a hierarchical vascular structure in warm blooded organisms. The self-organizing, self-remodeling, nature of vascular morphogenesis makes it a suitable platform to study the relative importance of different fundamental mechanisms of communication between cells. Two schools of thought have emerged regarding the fundamental guidance mechanisms in vascular morphogenesis: chemical soluble signaling and mechanical signaling. However, the systems used in the studies associated with these propositions have limited relevance as they do not recapitulate physiological conditions for important cellular events that take place during *in vivo*

vascular morphogenesis. In this sense, we argue that progress on the outstanding questions requires new approaches and tools.

In this work, I have studied the behavior of individual cells during vasculogenesis in a 3D collagen environment, as well as the characteristics of multi-cellular luminal invasions from a monolayer of ECs on a collagen scaffold during angiogenesis. The hypothetical framework is based on the reported ability of ECs to generate both chemical and mechanical signals and our methods exploit the random placement of cells as well as the structural heterogeneity of the culture systems developed. I then extend my efforts toward an application of directed vascular growth in scaffolds for tissue engineering.

One of the most challenging limitations in tissue engineering to date is the proper oxygenation and nourishment of the thick tissues of interest. Specifically, growing pre-vascularized tissues requires fluid access to capillary-sized vessels whose inlets are not spatially distributed in a controllable fashion. I identified and exploited the interactions of ECs undergoing vasculogenesis with lumenized angiogenic invasions in a microfluidic setup. I report that this approach produces capillaries with continuous connections (anastomosis) between an EC-lined channel (which can be connected to a flow source) and capillary vessels within a tissue, providing extended access for nourishment into the bulk of a tissue culture.

BIOGRAPHICAL SKETCH

Anthony Diaz-Santana was born on August 30, 1983 in San Juan, Puerto Rico. He is the oldest of three. Anthony grew up in a conservative family and surrounded by his very big extended family. His grandparents took care of him after school hours. Anthony was a very active child and teenager, being involved in tennis, swimming, guitar lessons, singing, and sound technician (at his old church). Anthony always had a genuine interest in science and mathematics, but it was not until he enrolled in the University of Puerto Rico that he discovered his calling: Chemical Engineering. Throughout his undergraduate training, Anthony was an active individual outside of the classroom with a passion for helping others: he enrolled in undergraduate research; he was a Math and Chemistry tutor; for his school's chapter he acted as the Treasurer of the Puerto Rican Institute of Chemical Engineers and Vice-president of the American Institutes of Chemical Engineers; he got involved in the department's curricular revision process; he was recipient of several scholarships, including the Merck Engineering and Technology Fellowship, and the National Institutes of Health Undergraduate Scholarship Program (both included internships). Anthony graduated Magna Cum Laude in 2007 from his bachelor's and decided that Cornell University had the best opportunities for his graduate career. Anthony was awarded the Sloan Foundation fellowship for 3 years and started working with Abraham Stroock unraveling how blood vessels are formed *in vitro* and how we can use this knowledge for tissue engineering. Anthony defended his PhD in Chemical and Biomolecular Engineering with a job under his belt with L'Oreal Research and Innovation in Clark, NJ. His scientific expertise will now grow in the field of surfactant and emulsion science for photoprotection technologies.

*Dedico este esfuerzo a mi bella madre Lourdes Santana Nieves y a mi abuelo singular,
Angel Luis Díaz Pérez; son los pilares de mi existencia.*

ACKNOWLEDGEMENTS

I would like to thank Abe for his guidance, support, and teaching me to always make my talks like a story; he holds too much information in his brain and it is impressive. Thank you Cynthia Reinhart-King and William Olbricht for getting on board as my committee members and your insightful feedback.

Thanks to my family who show unconditional support and love: Noah O’Neil Santiago Díaz (nephew), Andrea Sofia Santiago Díaz (niece), Stephanie Díaz Santana (sister), Nathalie M. Díaz Santana (sister), Lourdes Santana Nieves (mother), Jorge L. Díaz Acevedo (father), Orland Cruz Ramos (stepfather), Inés Acevedo Rodríguez (grandmother), and Angel L. Díaz Pérez (grandfather).

Thanks to my friends for being there with ears, arms, and partying with me: Jeisa Pelet (former student in Putnam lab), Alexis Torres, Carlos Dorta-Quiñones (former roommate), Ronald Ilma (former roommate), Henrique Aveiro (former roommate), Karen Osorio, Mariela Ferrer Zequeira, Edward Camacho (CNF), Pedro Meléndez, César Rivadeneyra, Heidi Rivadeneyra, Danielle D’Ambrosio, Juan Daniel Valderrama (former student in DeLisa lab), Orlando Ferrer, Lemuel Resto, John Morgan (also labmate), Vinay Pagay (also labmate), Karla Martínez, Anabel Piñeiro and Max Kellman.

Thanks to my former labmates for accepting that I can dance while I work in the lab and your helpful discussions: John Morgan, Lina Aboulmouna, Dr. Valerie Cross, Dr. Nakwon Choi, Dr. Ying Zheng, Dr. Michael Craven, Dr. Rachel Liang, the rest of the Stroock group, the Wu group, and the Fishcbach group.

Forever in debt.

TABLE OF CONTENTS

Biographical sketch	iii
Dedication	iv
Acknowledgements	v
Chapter 1 Introduction	1
Chapter 2 Investigation of the Interactions that Guide Invasion Angiogenesis and Vasculogenesis In Vitro	26
Chapter 3 Engineering Cellular Interactions to Drive Self-Assembling of Microvasculature In Vitro	71
Chapter 4 Conclusions and Future Directions	106
Appendix A Experimental Protocols	117
Appendix B Modeling the Crystallization of Hydroxyapatite in a Double Diffusion System	134
Appendix C Supplementary Figures for Chapter 2 and Chapter 3	160
Appendix D Matlab Codes	168

CHAPTER 1

INTRODUCTION

1-1 MORPHOGENESIS AND ORGANIZED STRUCTURES IN BIOLOGY

The organization of multi-cellular structures takes place early on during embryogenesis. Specific interactions between cells of a single type (homotypic) and of different cell types (heterotypic) inside a tissue give rise to functionality. As a familiar example, the epithelium (lining the cavities and surfaces of structures such as organs) gives rise to a permeability barrier between the external and internal environments of the tissue lined due to the mechanisms of adhesion and junctions along the polarization line of individual polarized (basal to apical) cells [1]. The intricate transfer of information in neural networks is possible due to the structure of the neuron: (1) small processes – dendrites – receive synaptic signals into the cell body (containing the nucleus of the cell), (2) a long axon conducts the signal away from the cell body, and (3) terminal branches emit the neuronal signal to many other target neurons [2]. The highly elongated skeletal muscle tissue cells are a syncytium of many individual cells fused together sharing a common cytoplasm; the basic contractile units of all cells reside in the cytoplasm and are responsible for skeletal motion upon activation [2]. Of interest to me in this thesis, individual endothelial cells (cell type that lines the insides of blood vessels, ECs) connect with neighboring ones forming multi-cellular tubular network structures that allow the transport of blood. The cellular orientation and organization in the examples described above lead to a tissue structure and function.

Understanding how global organization emerges from individual units is an outstanding challenge in the field of developmental biology [1]. Self-organization spans different fields, such as chemistry [3], economics [4], and biology [5]. To identify a system as a self-organizing the individual units must: interact with one another, be able to readjust their position in order to form a structure, and must be physically dynamic to explore their local environment. We classify blood vessel morphogenesis as a self-organization process given that both *in vivo* and *in vitro*: (1) individual ECs interact within a system that initially contains them individually distributed [1] to form a highly connected network of vessels, (2) ECs are motile (including migration) within their local environment, (3) they explore their local environment extending protrusions into the extracellular space (whether these protrusions are from individual cells or multi-cellular structures), and (4) no phenotypic pre-patterning exists in the environments in which they are embedded. We argue that the signals that generate interaction of individual cells occur at a local scale in vascular morphogenesis. We aim to understand the fundamental mechanism of communication of individual ECs by working within a controlled environment, *in vitro*.

Endothelial cells reorganize themselves creating vacuolar (matrix-free) spaces which eventually fuse into a larger luminal conduit – a process called vasculogenesis [2, 6-7]. During vasculogenesis individual cells connect with neighboring ones forming multi-cellular tubular network structures. The morphological changes of individual endothelial cells (ECs) and the interactions between multiple cells are important features in the blood carrying function of a vessel: delivering nutrients and oxygen. In addition, an endothelium (the collection of endothelial cells that form a single lumenized vessel) interacts with perivascular cells (ie. Pericytes, Smooth Muscle Cells) that contribute to vessel stability and to interactions of endothelium with tissue cells. The formation of networks of vessels from individual cells is a process regarded as self-

organizational whereby individual cellular behavior dictates the emergence of the global morphology of the network structure of vessels. In sections 2 through 4, I will give a brief overview of vascular development and physiology. Section 5 contains the state of the art relevant to cell communication during vascular development. In section 6, I present a brief overview of the applications of studying vascular development for tissue engineering.

1-2 BLOOD VESSELS

The vascular system supports the transport of nutrients and oxygen, the removal and transfer of metabolic products, the distribution of hormones in many organisms and cellular transport) [8]. There are three main classes of blood vessels in the vascular tree: arteries, veins, and capillaries. These are subdivided into transitional segments: arterioles (from arteries to capillaries) and venules (from capillaries to veins). While their physiological roles are distinct, the different blood vessels share basic characteristics that give them an intrinsic carrier capacity: blood vessels have a hollow space, called lumen, which is roughly in the shape of a cylinder along an individual segment of a vessel. The lumen is lined by a single layer of endothelial cells (ECs) [2] (**Fig. 1**). Arteries and veins are the largest blood vessels and act as a convective system. They have a thick, elastic wall of connective tissue and many layers of smooth muscle cells (SMCs) [2]. The amount of SMC layering and connective tissue vary with vessel diameter and function. However, in the smallest branches of the vascular tree, where we find capillaries, the walls consist of ECs and a basal lamina that is composed of extracellular matrix (ECM) proteins [2] with sparse pericytes around them. It is through the capillaries that form the “microcirculation” that the vascular system performs its essential functions [1], namely diffusive exchange with living tissue.

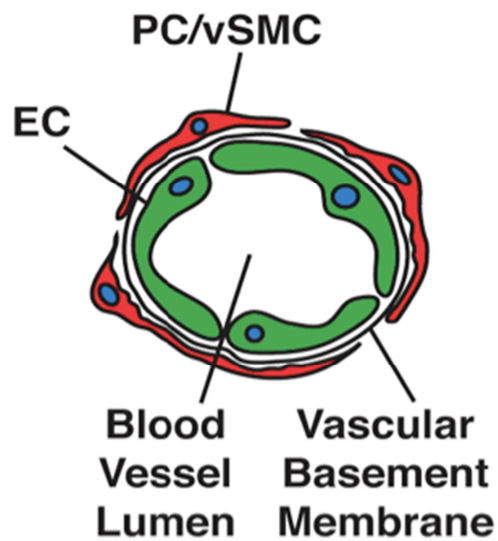


Figure 1. Schematic representation of a blood vessel. Depicted are the vessel lumen, capillary lining of ECs (green), and wall components (pericyte/SMC, red). Blue: cell nuclei. (Bergers and Song, 2005; [9]).

1-3 DEVELOPMENT OF THE VASCULATURE

The vasculature of the developing embryo establishes itself before the heart starts to pump and, rather than sprouting from the heart, the blood vessels form independently and link to the heart soon afterward [1]. The development of the vasculature is divided into two sequential steps: vasculogenesis and angiogenesis [10]. Vasculogenesis refers to the *de novo* formation of capillaries from individual ECs, while angiogenesis refers to new blood vessel formation from preexisting capillary vessels [6, 10-14]. Associated growth and mitogenic (leading to mitosis) factors with their associated receptors, regulate these processes. Although the specific role for relevant factors involved in vascular morphogenesis is still under active investigation, vascular endothelial growth factor (VEGF) is the most prominent of these.

1-3.1 Vasculogenesis

Vasculogenesis is subdivided into three phases, *in vivo*. The first phase consists of the differentiation of splanchnic mesodermal cells into hemangioblasts [15-16]. The mesoderm is the germ layer of the embryo that gives rise to the blood, heart, kidney, gonads, bones, and connective tissue [1]. Hemangioblasts on the other hand, are the precursors of both the blood cells and the blood vessels [16], and they condense into aggregations named blood islands [7, 11, 15-16]. The inner cells of these islands become hematopoietic stem cells (precursors for all blood cells), and the outer cells become angioblasts (precursors of ECs) [1, 7, 11]. In the second phase of vasculogenesis, angioblasts multiply and differentiate into ECs in the yolk sac [1]. In the third phase, the ECs form tubes (tubulogenesis [6]) and connect to form the primary capillary plexus [1, 7, 11]. Two receptors for VEGF are expressed by angioblasts and by ECs. VEGF-receptor-2 (VEGFR-2) and VEGFR-1 are upregulated during vasculogenesis and angiogenesis. When mice

are disrupted in VEGFR-2 gene expression vasculogenesis does not take place [17] whereas if VEGFR-1 is absent angiogenesis is prevented [18].

After angioblasts have become ECs, they undergo morphogenic remodeling to form the lumen and a tube-like structured network of vessels as illustrated in **Fig. 2**. Although it is known the ECM has significant effects on ECs, the molecular mechanisms implicated in role of ECM on morphogenesis had been unknown until recently. Lumens formed by ECs seeded in 3D collagen matrices resemble tube structures in collagenous environments *in vivo*. In this system (*in vitro*), Davis and colleagues showed that capillary morphogenesis was regulated via the ECM receptor *integrin $\alpha_2\beta_1$* [19], which is induced in ECs by VEGF [20]. Collagen type I interaction with integrins $\alpha_2\beta_1$ and $\alpha_1\beta_1$ suppresses the activity of cyclic-AMP and protein kinase A leading to the generation of actin stress fibers and EC contractility (implicated in tube formation). After ECs have undergone morphogenesis they are not physiologically functional unless the newly formed capillary network is stabilized. For this, ECs coordinate extracellular signaling that recruits pericytes (sometimes identified as SMCs) with upregulation of ECM protein secretion that will compose the basement membrane (BM) and affect its local environment cues. One example of local effect on EC morphogenesis is the production of the BM proteins laminin and collagen type IV that promote vascular stabilization by isolating the vessels from collagen I. The morphogenesis and quiescence starts are alternate during angiogenesis as I will develop further within the next section.

1-3.2 Angiogenesis

Angiogenesis is the process in which the primary plexus is remodeled and pruned into a distinct capillary bed, arteries, and veins [1, 6, 10-11, 14, 21] (**Fig. 3**). It also involves the

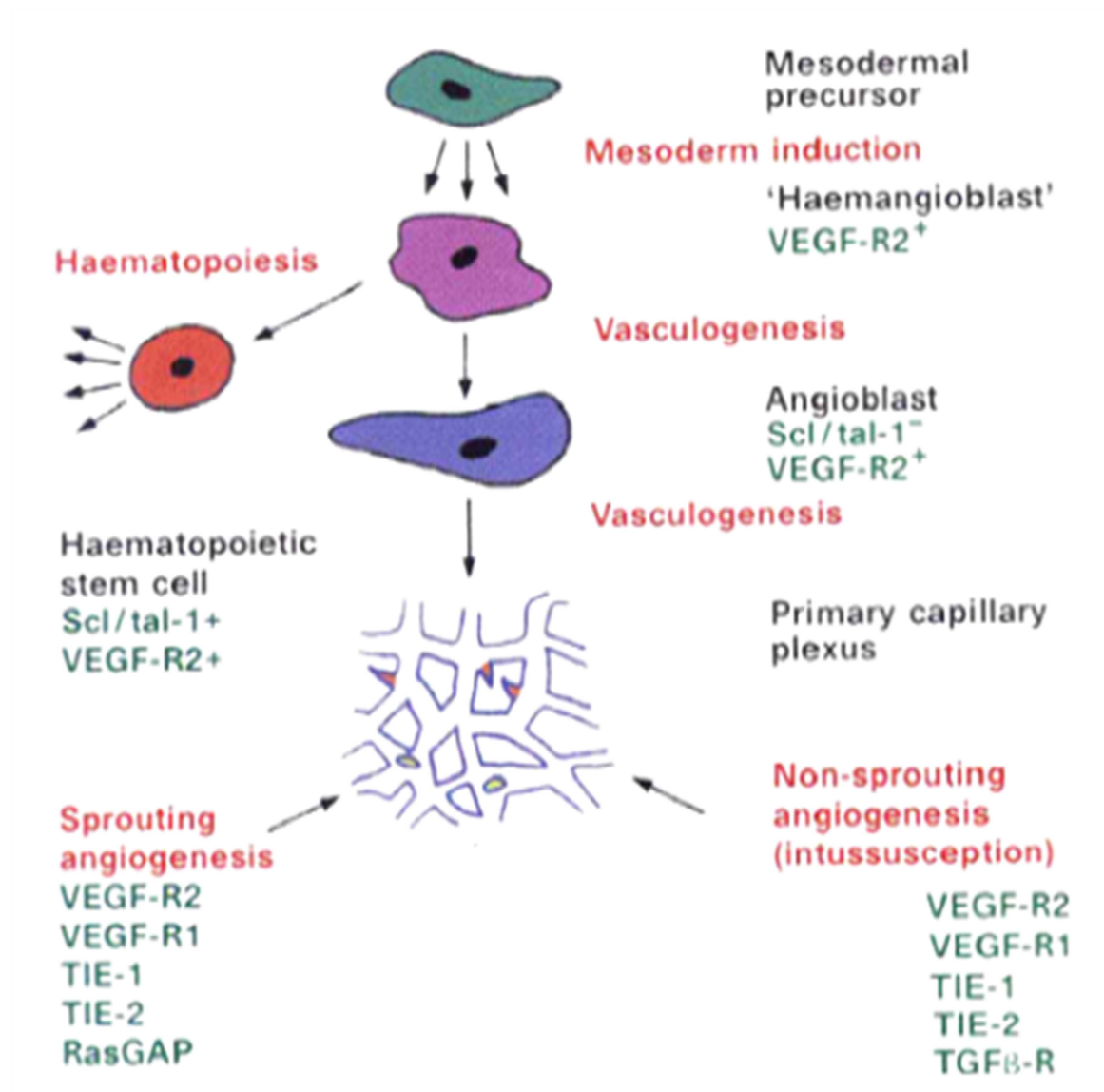


Figure 2. Vascular development flow chart. Processes (red labels), molecules (green labels) and appearances (black labels) taking place in vascular development. Red tips are sprouts and green circles are non-sprouting segments. This is a schematic from observations seen in the avian yolk-sac vascular system. (Adapted from Risau, 1997; [11]).

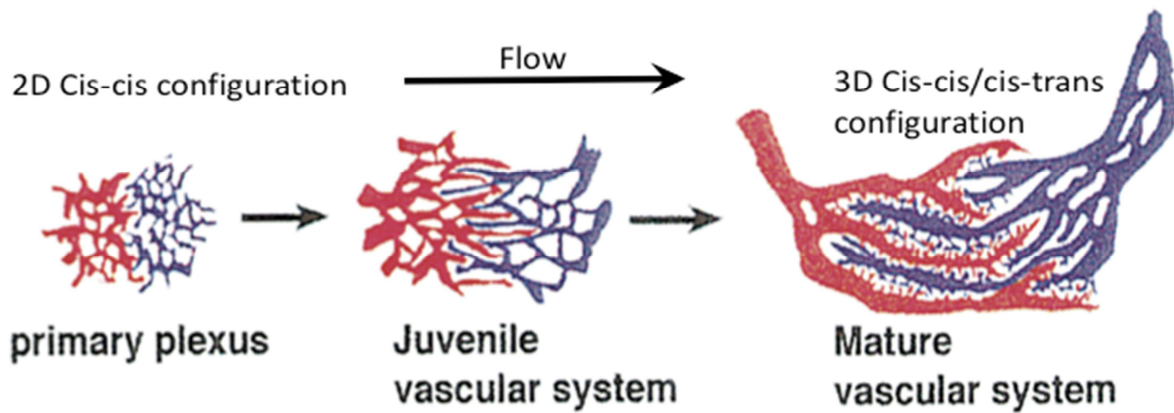


Figure 3. Maturation of capillary plexus. The original two-dimensional (2D) capillary plexus contains two subpopulations of ECs, arterial ECs (red) and venous ECs (blue). Flow initiates morphogenic changes and capillary maturation. Remodeling induces the plexus to become a 3D network with interdigitating veins and arteries. Transitions from arterial to venous cells occur through capillaries. (Adapted from Wang, et al 1998; [13])

differential recruitment of associated supporting cells to different segments of the vasculature [21]. Although angiogenesis is yet to be fully understood, observations suggest that there are at least two types of angiogenesis: sprouting of capillaries from pre-existing vessels, and non-sprouting angiogenesis (longitudinal division of existing vessels [22]) [11]. Sprouting angiogenesis involves (1) the degradation of ECM followed by (2) migration and proliferation of ECs, (3) formation of a lumen, and (4) functional maturation of the endothelium [11]. Degradation of ECM activated by paracrine (signaling from a cell type that is different from target cell type) and autocrine VEGF signaling is achieved by ECs through the matrix metalloproteinases (MMPs). Membrane type-MMPs have a central role in angiogenesis since invasion of ECs must be preceded by the degradation of the laminin-rich BM [23]. The ECM promotes EC migration and proliferation – once the ECs degrade the BM – through cell-surface integrins by the activation of the mitogen-activated protein kinase (MAPK) [23]. ECM interactions are equally important for EC survival since MAPK signaling pathways suppresses cell death [23]. Morphogenesis reassumes once the ECs are no longer surrounded by BM. The stability of the vasculature is regained by the reestablishment of BM and the recruitment of the smooth-muscle related pericytes. Pericytes are embedded within the BM of the microcirculation and contribute to its generation [9]. Pericytes in capillaries extend long processes parallel to the long axis of the vessel as well as smaller processes circumferentially [24]. In addition to physically supporting ECs by loosely sheathing them, pericytes communicate with ECs through gap junctions (which enable the transfer of ions and small molecules between the different cell types) and paracrine signaling [9]. Pericyte-EC interactions originate by the pericyte recruitment associated with the secretion of the platelet-derived growth factor-BB homodimer (PDGF-B) by ECs [25]. The EC-pericyte interaction is crucial for the subsequent maturation process. However,

functional maturation of the endothelium, more significantly, relies on an essential physiological process: *blood circulation*.

1-3.3 Maturation of Capillary Plexus: the Role of Physical Forces

During angiogenesis, the capillary plexus is remodeled to resemble a mature system with larger and smaller vessels [11] in order to develop a functional vascular loop and accommodate cardiac output [8]. One of the hallmarks of remodeling the capillary plexus is the differentiation of arteries and veins. Until recently, vascular biologists assumed that the immature capillary plexus was a homogeneous cell type network of ECs [21] and that differentiation was achieved later in embryonic development under hemodynamic forces. However, Anderson and colleagues [13] demonstrated in the mouse embryo that, prior to perfusion, there is a subset of ECs in the capillary plexus that expresses the transmembrane protein ephrin-B2, and another subset, the transmembrane protein Eph-B4 (the receptor for ephrin-B2) (**Fig. 3**, primary plexus); these markers served to determine the fate of the ECs since arterial cells expressed ephrin-B2 whereas venous cells Eph-B4 [13]. Thus, the capillary plexus shows a polarity of protein expression where arterial ECs are in the posterior side of the embryo and veins occupy the anterior side (**Fig. 3**). Initially, both populations are interconnected in a 2D *cis-cis* conformation (flow passes continually from an arterial EC to a venous EC) and after the onset of flow it is remodeled into a 3D *cis-cis* and *cis-trans* configuration (in *cis-trans* flow is not passed from arterial to venous ECs but the cells may be interacting through their abluminal surfaces) (**Fig. 3**) [21]. The information gained from their experiments suggested that arterial-venous fate may be genetically predetermined. Remodeling via angiogenesis, however, does not take place until the vasculature is perfused by blood flow (**Fig. 4**). Studies in the chick yolk sac led by Eichmann [8] have established that ephrin-B2/Eph-B4 proteins serve as markers instead of makers of arteries and

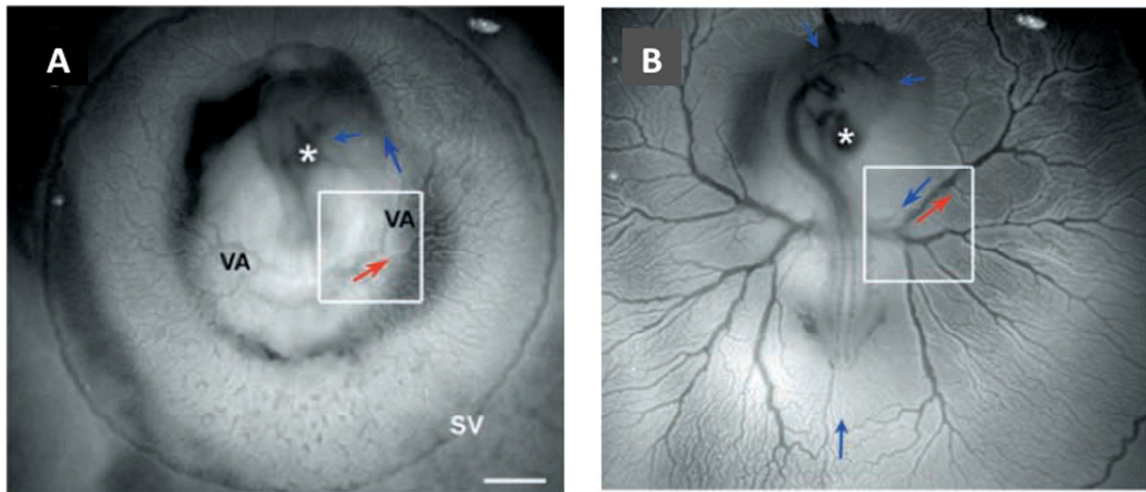


Figure 4. Remodeling of vessel network through perfusion. (A) Chick embryo yolk sac prior to perfusion. (B) Embryo 26 hours later. Blood flow has started and remodeling of the vasculature takes place. Red: arterial flow; blue: venous flow. VA: vitelline artery. Asterisk shows the heart (Adapted from le Noble, 2004; [8])

veins. In their work, it was shown that blood flow changes the expression patterns of the ephrin family proteins. Arterial ECs maintain their ephrin-B2 expression if perfused and cease to do so if not. Furthermore, EC plasticity is required for remodeling in that ECs can change identity. Arterial ECs disconnected from arterial segments, lost arterial expression, and were subsequently reconnected into the growing venous capillary plexus [8]. This work represented a prominent contribution to our understanding of the vascular development since it is now clear that genetics can couple with physical forces (flow) to change a genetic predetermination. There are many more relevant reasons why blood flow is important.

Blood flow carries nutrients, oxygen, and signaling molecules to the vessels as well as creates physical forces on the ECs of the vessel wall [26]. Any of these cues could have an effect on the remodeling of the vasculature in the early embryo. In a recent work, Dickinson and coworkers distinguished the requirement of hemodynamic force for proper remodeling [27]. When the heart starts to beat it flows plasma through the circulation and erythroblasts – a red blood cell (RBC) that still retains a nucleus and is the immediate precursor of RBCs – enter at a later stage [26]. RBCs are known to be transporters of oxygen and their presence in the blood also increases its viscosity substantially. In order to separate the chemical effects derived from oxygen enriching RBCs from the physical forces that would arise from the same RBCs by an increase in the apparent viscosity of plasma, they used mice in which erythroblasts were sequestered inside the blood islands, preventing vascular remodeling. With this setup, the team injected hetastarch (a high-molecular-weight synthetic sugar derived from plants) into the heart of the embryo artificially increasing the viscosity of blood and rescuing morphogenic changes. Their results suggest that convective transfer of signaling molecules and of oxygen is not necessary for remodeling since embryos with low hematocrit (volume fraction occupied by

RBCs) showed no maturation of their plexuses upon perfusion [27]. However, oxygen is still dissolved in the flowing plasma (but as a much lower level than with RBCs present); hetastarch may function as a source of energy for metabolism to be carried on. These experiments further suggest that shear stress is responsible for morphogenic changes seen in the capillary plexus given that its rescued remodelability depended on shear. This response points to the well-established concept that ECs have a mechanosensing role. Elucidation of the molecular mechanisms underlying this mechanosensing remains an active area of research. For example, Tzima and coworkers gave mechanoresponsiveness to normally unresponsive cells through transfection with platelet/endothelial cell adhesion molecule-1 (PECAM-1), vascular endothelial cell cadherin (VE-cadherin), and VEGFR2; this study suggests that the expression of this trio of proteins plays role in mechanosensing in ECs [28]. However, VE-cadherin showed no dependence on fluid force in mice and PECAM-1-knockout mice are viable to adulthood [27]; which points to the complexity underlying the identification of the molecular agents responsible for the mechanosensitivity of ECs *in vivo*. An *in vitro* system that mimics physiological events would enhance the understanding of mechanoreception in ECs, vessel growth, effects of flow conditions, and potentially impact the tissue engineering community by studying these events in a relative isolation.

1-4 CELL COMMUNICATION WITH MICROENVIRONMENT

Cell communication (mechanism that leads cells to interact) can take place *at a distance* or by direct *physical contact* (**Fig. 5**). Individual cells generate signals that dissipate in the extracellular space: soluble chemical signals secreted by cells or mechanical signals generated due to the traction on extracellular matrix (ECM) generated by cell motility via diffusion

(chemical) or relaxation (mechanical). Cells can secrete soluble signals in the form of growth factors (e.g. VEGF, FGF, EGF), proteinases (e.g. metalloproteinases), chemokines (e.g. IL-8), and small molecules (e.g. nitric oxide) [2]. These secretions can generate concentration gradients which cells are able to respond to in the form of cell growth, proliferation, and migration [1-2, 29]. Biomolecular gradients play an important role in biological processes such as development [30-32], cancer metastasis [33-34], and inflammation [35-36]. *In vitro* experiments have also shown that cells under externally imposed gradients of biomolecules can orient themselves [37] and also exhibit chemotaxis (migration along a gradient of chemokines) [38-39]. Recently, it has been shown that individual cells can respond to traction forces generated by neighboring cells [40-42]. Contact-dependent signals may take place by agonistic or antagonistic relationship between proteins on the cell membrane with: (1) proteins on a neighboring cell's membrane, (2) anchored on matrix fibers, or (3) with proteins that compose a matrix. For instance, the ephrin A1 protein on a signaling cell's membrane comes in contact with EphA4 protein on the membrane of a target cell regulating the migration of the growth cone in neurons [43]. Endothelial cells exhibit different response to VEGF when VEGF is matrix-bound compared to when it is diffusible [44]. Fibroblasts are found to exhibit a haptotactic response (migration along a gradient of cellular adhesion sites) towards fibronectin (FN) gradients [45].

Specifically for vascular networks, pattern formation using guidance cues that act at a distance has been identified *in vivo* [46]. Gerhardt, et. al. reported that VEGF gradients control the angiogenic sprouting of ECs that overlay astrocytic networks by guiding filopodial extensions of the tip cell [32]. In addition, contact-dependent signaling via Delta4-Notch1 signaling has also been reported in the vasculature [47]. *In vitro*, different approaches have been used to characterize how ECs respond to their environmental cues [48]. For instance, Reinhart-

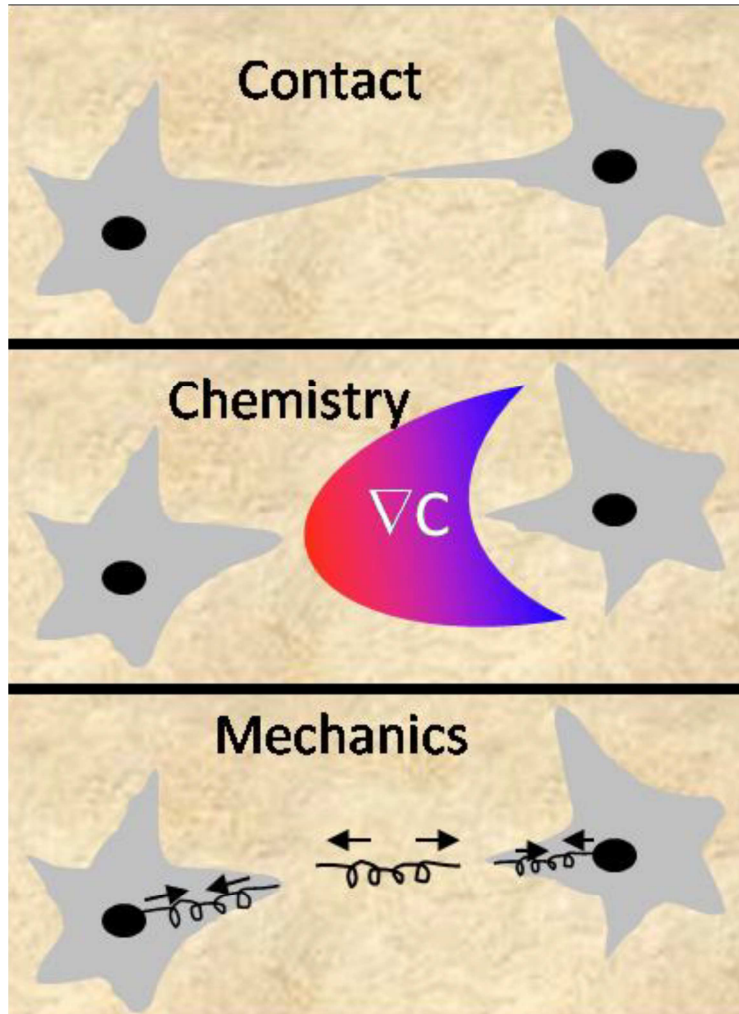


Figure 5. Modes of cell communication. Cell-cell contact, chemical gradients, and mechanical interactions have all been implicated as possible mechanisms of cell-cell communication during vascular network formation. The similar characteristics of signal decay via of mass and stress transfer within a 3-D matrix make decoupling these mechanisms difficult. (Valerie Cross, 2010; [49])

King et al. reported that ECs are capable of sensing mechanical forces exerted on substrates by adjacent cells and that this sensing mechanism hinders the migration of cells in pairs promoting the formation of cell-cell contacts [40]. This work suggested that cells in pairs can sense each other at a separation distance on the order of cell size. Manoussaki et. al, proposed a mechanical model whereby cellular networks form as a result of traction forces exerted by the cells on the matrix, and suggests that cells do not need to migrate for network formation [50]. In contrast, Serini et al., suggests that cell migration is directed by the presence of gradients of secreted VEGF in an autocrine fashion, and that this migration (on the order of a cell diameter) leads to interconnected networks [51]. Yin et. al. report that EC migration speed and migration patterns were strongly affected by secreted collagen and VEGF when pairwise EC-EC interactions were assessed [52]. Furthermore, Shamloo et al, reports that VEGF gradients can induce EC polarization and chemotactic migration [37].

One of the major challenges in studying cell-cell communication *in vivo* is the concurrent extracellular signals received by an individual cell inside a complex network of cell systems. In **Chapter 2**, my efforts towards understanding cell communication during vascular morphogenesis in three-dimensional (3D) *in vitro* systems are discussed. The considerations in **Chapter 2** are on the scale of cell sizes. In the following section, I briefly overview the state of the art for tissue engineering approaches; this topic pertains to the macroscopic distances that need to be considered for vascularization of living tissues (which I later discuss in **Chapter 3**),

1-5 MIMICKING LIVING VASCULATURE IN VITRO: TISSUE ENGINEERING

APPROACHES

Achieving a perfused (flow through luminal spaces) *in vitro* network of blood vessels has been a long standing goal within the tissue engineering community due to its implications in the growth of replaceable tissues. In addition, such a platform could open new doors to address unanswered questions in vascular development and tissue diseases such as tumor angiogenesis [53]. Neumann and colleagues first attempted to create a perfused vessel *in vitro* consisting of a nylon thread that supported smooth muscle cells (SMCs) on the surface until they reached confluency and the nylon could be removed; flow was introduced after the removal of nylon [54]. In this system, ECs were not present. Later, Chrobak [55] and concurrently Frerich [56], created collagen scaffolds that presented a single empty cylinder where ECs were seeded to form an endothelium. Flow was introduced into these systems to study the permeability of endothelium [55] and to assess the capacity of ECs lining the collagen tube to invade from a vessel monolayer [56]. These first attempts at growing blood vessels in culture opened the doors to more sophisticated ideas for tissue vascularization and the consideration of perfusion of such vessel structures.

Vascularizing tissues *in vitro* requires the construction of cellular structures over macroscopic distances. The efforts to accomplish this goal include: tailoring geometrical features on acellular biomaterials that accelerates cellular transport from host tissue into the implanted tissue [57], taking advantage of the self-organization of ECs or EPCs into vascular networks (vasculogenesis) in 3D matrices, establishing co-cultures with specific tissue cell [58-61], and intricate microfluidic path geometries to line collagen walls with ECs [62-65] (elegant progressions of Chrobak [55], and Frerich [56]). The works of Levenberg et. al.[58], and Koike, et. al.[61], deserve particular attention since both showed that when proper vascular tubes were grown in 3D matrices, the vasculature in these constructs connected (anastomosis) with host

vasculature and became functional after they have been implanted into mice. However, the size and cell densities of these tissue constructs (~ 1 mm thick, 3.2×10^7 cells/mL [66]) limits the generalization of this effort to engineering thick tissues. These design parameters restricted the dimension of constructs to the order of their Krogh length (the distance over which oxygen can diffusive before being consumed; ~ 1 mm) [67-68] due to the lack of perfusion during their culture *in vitro* and post-implantation before anastomosis. In order to scale this effort to create highly cellularized and or thick tissues we need to establish a connection between an external source of blood or growth medium and a developing 3D capillary bed *in vitro*, before implantation. However, when vascular networks are created *in vitro* by vasculogenesis, the locations of vessel ends (vessel openings) are not controlled. This issue presents a challenge to the design of systems that provide connections between the lab-scale and the capillary vessels in the bulk of the tissue construct. Anastomosis represents the native mechanism to generate such connections *in vivo*, and could be exploited *in vitro*.

Anastomosis is still a poorly understood process in the developing vasculature. We know that tip cells of angiogenic sprouts migrate into open tissues (angiogenesis) and that signals such as vascular endothelial growth factor (VEGF) play a crucial role in their guidance [32]. The angiogenic exploration of the tip cells guides stalk cells from the pre-existing blood vessel into the adjacent tissue; upon encountering other vessels, anastomosis can occur. However, we do not understand what an individual sprout does when it encounters another vessel. Song et. al. recently developed a PDMS microfluidic platform to assess the mechanisms that govern anastomosis [69]. They reported that *in vitro* angiogenic sprouts originating from opposing monolayers of ECs are able to anastomose leading to perfuseable vessel formation. This perfuseability of the vasculature *in vitro* is a highly desired feature for tissue engineering

systems. The size and material choice of their system remains to limit its applicability to growing tissues: (1) the health of a tissue could be compromised if a large tissue is studied using the same approach given that the vasculature would require a long time for anastomosis, (2) the remodelability that an evolving vasculature exhibits is hindered by the PDMS that forms the device, and (3) PDMS is not a biocompatible material suitable for implantation.

In **Chapter 3**, I describe the generation of a platform to study anastomosis of an invading monolayer of ECs and an underlying bed of blood vessels generated by vasculogenesis. Then, I describe how this concept can be extrapolated into a microfluidic context in order to create an intricate interconnected network of vessels with just a few flow inlets.

1-6 REFERENCES

1. Gilbert, S.F., *Developmental Biology*. 6 ed. 2000.
2. Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., Walter, P., ed. *Molecular Biology of The Cell*. 5th ed. 2008, Garland Science.
3. Love, J.C., et al., *Self-assembled monolayers of thiolates on metals as a form of nanotechnology*. Chem Rev, 2005. **105**(4): p. 1103-69.
4. Witt, U., *Self-organization and economics - what is new?* Structural Change and Economic Dynamics, 1997. **8**(4): p. 489-507.
5. Strong, M., *Protein nanomachines*. PLoS Biol, 2004. **2**(3): p. E73.
6. Davis, G.E., K.J. Bayless, and A. Mavila, *Molecular basis of endothelial cell morphogenesis in three-dimensional extracellular matrices*. Anatomical Record, 2002. **268**(3): p. 252-275.
7. Risau, W. and I. Flamme, *Vasculogenesis*. Annual Review of Cell and Developmental Biology, 1995. **11**: p. 73-91.
8. le Noble, F., et al., *Flow regulates arterial-venous differentiation in the chick embryo yolk sac*. Development, 2004. **131**(2): p. 361-75.
9. Bergers, G. and S. Song, *The role of pericytes in blood-vessel formation and maintenance*. Neuro Oncol, 2005. **7**(4): p. 452-64.
10. Hanahan, D., *Signaling vascular morphogenesis and maintenance*. Science, 1997. **277**(5322): p. 48-50.
11. Risau, W., *Mechanisms of angiogenesis*. Nature, 1997. **386**(6626): p. 671-674.
12. Folkman, J. and P.A. D'Amore, *Blood vessel formation: what is its molecular basis?* Cell, 1996. **87**(7): p. 1153-5.
13. Wang, H.U., Z.F. Chen, and D.J. Anderson, *Molecular distinction and angiogenic interaction between embryonic arteries and veins revealed by ephrin-B2 and its receptor Eph-B4*. Cell, 1998. **93**(5): p. 741-53.

14. Bayless, K.J. and G.E. Davis, *The Cdc42 and Rac1 GTPases are required for capillary lumen formation in three-dimensional extracellular matrices*. J Cell Sci, 2002. **115**(Pt 6): p. 1123-36.
15. Shalaby, F., et al., *A requirement for Flk1 in primitive and definitive hematopoiesis and vasculogenesis*. Cell, 1997. **89**(6): p. 981-90.
16. Cortes, F., et al., *Differential expression of KDR/VEGFR-2 and CD34 during mesoderm development of the early human embryo*. Mech Dev, 1999. **83**(1-2): p. 161-4.
17. Shalaby, F., et al., *Failure of blood-island formation and vasculogenesis in Flk-1-deficient mice*. Nature, 1995. **376**(6535): p. 62-6.
18. Fong, G.H., et al., *Role of the Flt-1 receptor tyrosine kinase in regulating the assembly of vascular endothelium*. Nature, 1995. **376**(6535): p. 66-70.
19. Davis, G.E. and C.W. Camarillo, *An alpha 2 beta 1 integrin-dependent pinocytic mechanism involving intracellular vacuole formation and coalescence regulates capillary lumen and tube formation in three-dimensional collagen matrix*. Exp Cell Res, 1996. **224**(1): p. 39-51.
20. Senger, D.R., et al., *Angiogenesis promoted by vascular endothelial growth factor: regulation through alpha1beta1 and alpha2beta1 integrins*. Proc Natl Acad Sci U S A, 1997. **94**(25): p. 13612-7.
21. Yancopoulos, G.D., M. Klagsbrun, and J. Folkman, *Vasculogenesis, angiogenesis, and growth factors: ephrins enter the fray at the border*. Cell, 1998. **93**(5): p. 661-4.
22. Conway, E.M., D. Collen, and P. Carmeliet, *Molecular mechanisms of blood vessel growth*. Cardiovasc Res, 2001. **49**(3): p. 507-21.
23. Davis, G.E. and D.R. Senger, *Endothelial extracellular matrix - Biosynthesis, remodeling, and functions during vascular morphogenesis and neovessel stabilization*. Circulation Research, 2005. **97**(11): p. 1093-1107.
24. Allt, G. and J.G. Lawrenson, *Pericytes: cell biology and pathology*. Cells Tissues Organs, 2001. **169**(1): p. 1-11.

25. Hagedorn, M., et al., *VEGF coordinates interaction of pericytes and endothelial cells during vasculogenesis and experimental angiogenesis*. Dev Dyn, 2004. **230**(1): p. 23-33.
26. Jones, E.A., F. le Noble, and A. Eichmann, *What determines blood vessel structure? Genetic prespecification vs. hemodynamics*. Physiology (Bethesda), 2006. **21**: p. 388-95.
27. Lucitti, J.L., et al., *Vascular remodeling of the mouse yolk sac requires hemodynamic force*. Development, 2007. **134**(18): p. 3317-26.
28. Tzima, E., et al., *A mechanosensory complex that mediates the endothelial cell response to fluid shear stress*. Nature, 2005. **437**(7057): p. 426-31.
29. Keenan, T.M. and A. Folch, *Biomolecular gradients in cell culture systems*. Lab Chip, 2008. **8**(1): p. 34-57.
30. Krieg, P.A. and O. Cleaver, *VEGF mediates angioblast migration during development of the dorsal aorta in Xenopus*. Development, 1998. **125**(19): p. 3905-3914.
31. Nagel, M., et al., *Guidance of mesoderm cell migration in the Xenopus gastrula requires PDGF signaling*. Development, 2004. **131**(11): p. 2727-2736.
32. Betsholtz, C., et al., *VEGF guides angiogenic sprouting utilizing endothelial tip cell filopodia*. Journal of Cell Biology, 2003. **161**(6): p. 1163-1177.
33. Eccles, S.A., *Targeting key steps in metastatic tumour progression*. Curr Opin Genet Dev, 2005. **15**(1): p. 77-86.
34. Strieter, R.M., et al., *CXC chemokines in angiogenesis of cancer*. Semin Cancer Biol, 2004. **14**(3): p. 195-200.
35. Firtel, R.A. and C.Y. Chung, *The molecular genetics of chemotaxis: sensing and responding to chemoattractant gradients*. Bioessays, 2000. **22**(7): p. 603-15.
36. Wu, D., *Signaling mechanisms for regulation of chemotaxis*. Cell Res, 2005. **15**(1): p. 52-6.
37. Shamloo, A., et al., *Endothelial cell polarization and chemotaxis in a microfluidic device*. Lab on a Chip, 2008. **8**(8): p. 1292-1299.

38. Haessler, U., et al., *An agarose-based microfluidic platform with a gradient buffer for 3D chemotaxis studies*. Biomed Microdevices, 2009. **11**(4): p. 827-35.
39. Cheng, S.Y., et al., *A hydrogel-based microfluidic device for the studies of directed cell migration*. Lab Chip, 2007. **7**(6): p. 763-9.
40. Reinhart-King, C.A., M. Dembo, and D.A. Hammer, *Cell-Cell Mechanical Communication through Compliant Substrates*. Biophysical Journal, 2008. **95**(12): p. 6044-6051.
41. Califano, J.P. and C.A. Reinhart-King, *A Balance of Substrate Mechanics and Matrix Chemistry Regulates Endothelial Cell Network Assembly*. Cellular and Molecular Bioengineering, 2008. **1**(2-3): p. 122-132.
42. Reinhart-King, C.A., Dembo, M., Hammer, D.A., *Endothelial Cell Traction Forces on RGD-Derivatized Polyacrylamide Substrata*. Langmuir, 2003. **19**(5): p. 1573-1579.
43. Murai, K.K. and E.B. Pasquale, *'Eph'ective signaling: forward, reverse and crosstalk*. J Cell Sci, 2003. **116**(Pt 14): p. 2823-32.
44. Chen, T.T., et al., *Anchorage of VEGF to the extracellular matrix conveys differential signaling responses to endothelial cells*. J Cell Biol, 2010. **188**(4): p. 595-609.
45. Rhoads, D.S. and J.L. Guan, *Analysis of directional cell migration on defined FN gradients: role of intracellular signaling molecules*. Experimental Cell Research, 2007. **313**(18): p. 3859-3867.
46. Gerhardt, H., et al., *VEGF guides angiogenic sprouting utilizing endothelial tip cell filopodia*. Journal of Cell Biology, 2003. **161**(6): p. 1163-1177.
47. Hellstrom, M., Phng, L.K., Hofmann, J.J., Wallgard, E., Coultas, L., Lindblom, P., Alva, J., Nilsson, A.K., Karlsson, L., Galiano, N., Yoon, K., Rossant, J., Iruela-Arispe, M.L., Kalen, M., Gerhardt, H., Betsholtz, C., *Dll4 signalling through Notch1 regulates formation of tip cells during angiogenesis*. Nature, 2007. **445**: p. 776-780.
48. He, Q.F., et al., *Improved cellularization and angiogenesis using collagen scaffolds chemically conjugated with vascular endothelial growth factor*. Acta Biomaterialia, 2011. **7**(3): p. 1084-1093.

49. Cross, V., *Biophysical and Chemical Mechanisms of Cellular Organization and Dynamics during 3-D Vasculogenesis*, in *Biomedical Engineering*. 2010 Cornell University: Ithaca.
50. Manoussaki, D., et al., *A mechanical model for the formation of vascular networks in vitro*. *Acta Biotheoretica*, 1996. **44**(3-4): p. 271-282.
51. Serini, G., et al., *Modeling the early stages of vascular network assembly*. *Embo Journal*, 2003. **22**(8): p. 1771-1779.
52. Yin, Z.Z., et al., *Analysis of pairwise cell interactions using an integrated dielectrophoretic-microfluidic system*. *Molecular Systems Biology*, 2008. **4**.
53. Stroock, A.D. and C. Fischbach, *Microfluidic culture models of tumor angiogenesis*. *Tissue Eng Part A*, 2010. **16**(7): p. 2143-6.
54. Neumann, T., B.S. Nicholson, and J.E. Sanders, *Tissue engineering of perfused microvessels*. *Microvasc Res*, 2003. **66**(1): p. 59-67.
55. Chrobak, K.M., D.R. Potter, and J. Tien, *Formation of perfused, functional microvascular tubes in vitro*. *Microvasc Res*, 2006. **71**(3): p. 185-96.
56. Frerich, B., K. Zuckmantel, and A. Hemprich, *Microvascular engineering in perfusion culture: immunohistochemistry and CLSM findings*. *Head Face Med*, 2006. **2**: p. 26.
57. Zheng, Y., et al., *Microstructured templates for directed growth and vascularization of soft tissue in vivo*. *Biomaterials*, 2011. **32**(23): p. 5391-5401.
58. Levenberg, S., et al., *Engineering vascularized skeletal muscle tissue*. *Nature Biotechnology*, 2005. **23**(7): p. 879-884.
59. Germain, L., M. Remy-Zolghadri, and F. Auger, *Tissue engineering of the vascular system: from capillaries to larger blood vessels*. *Medical & Biological Engineering & Computing*, 2000. **38**(2): p. 232-240.
60. Fuchs, S., A. Hofmann, and C.J. Kirkpatrick, *Microvessel-like structures from outgrowth endothelial cells from human peripheral blood in 2-dimensional and 3-dimensional co-cultures with osteoblastic lineage cells*. *Tissue Engineering*, 2007. **13**(10): p. 2577-2588.

61. Koike, N., et al., *Tissue engineering: creation of long-lasting blood vessels*. Nature, 2004. **428**(6979): p. 138-9.
62. Vickerman, V., et al., *Design, fabrication and implementation of a novel multi-parameter control microfluidic platform for three-dimensional cell culture and real-time imaging*. Lab on a Chip, 2008. **8**(9): p. 1468-1477.
63. Moya, M.L., et al., *In vitro perfused human capillary networks*. Tissue Eng Part C Methods, 2013. **19**(9): p. 730-7.
64. Miller, J.S., et al., *Rapid casting of patterned vascular networks for perfusable engineered three-dimensional tissues*. Nat Mater, 2012. **11**(9): p. 768-74.
65. Zheng, Y., et al., *In vitro microvessels for the study of angiogenesis and thrombosis*. Proc Natl Acad Sci U S A, 2012. **109**(24): p. 9342-7.
66. Levenberg, S., et al., *Differentiation of human embryonic stem cells on three-dimensional polymer scaffolds*. Proc Natl Acad Sci U S A, 2003. **100**(22): p. 12741-6.
67. Choi, N.W., et al., *Microfluidic scaffolds for tissue engineering*. Nature Materials, 2007. **6**(11): p. 908-915.
68. Krogh, A., *The number and distribution of capillaries in muscles with calculations of the oxygen pressure head necessary for supplying the tissue*. J Physiol, 1919. **52**(6): p. 409-15.
69. Song, J.W., D. Bazou, and L.L. Munn, *Anastomosis of endothelial sprouts forms new vessels in a tissue analogue of angiogenesis*. Integrative Biology, 2012. **4**(8): p. 857-862.

CHAPTER 2

INVESTIGATION OF THE INTERACTIONS THAT GUIDE INVASION ANGIOGENESIS AND VASCULOGENESIS IN VITRO

2-1 INTRODUCTION

Cell-cell communication controls tissue structure and function, and enables tissue survival. An individual cell experiences myriad signals inside a tissue: contact-dependent signals mediated by membrane-membrane contact, diffusible endocrine signals, and extracellular matrix (ECM) interactions [1]. Contact-dependent signaling takes place when membrane-associated receptor-ligand signals are activated [2-3]; endocrine signals are mediated by factors secreted by cells into the extracellular environment that are processed by neighboring cells [4]; and cells can exert tractional forces on the ECM that are sensed by adjacent cells [5] (**Fig.1A**). One of the major challenges in studying cell-cell communication *in vivo* is the isolation of the concurrent extracellular signals received by an individual cell inside a tissue. To aid in this dissection, we turn to engineering microenvironments that allow for recapitulation of cellular events and biochemical parameters. This approach can simplify and enable the control of the experimental parameter space. Elucidating the guiding mechanisms for cell-cell communication may enable the rational design and manipulation of living systems from the cellular scale up to the tissue scale. In this work, we focus on understanding the communication mechanisms that endothelial cells (EC) use during the development of vascular networks; this study aims to advance toward the goal of creating healthy tissues, *in vitro*.

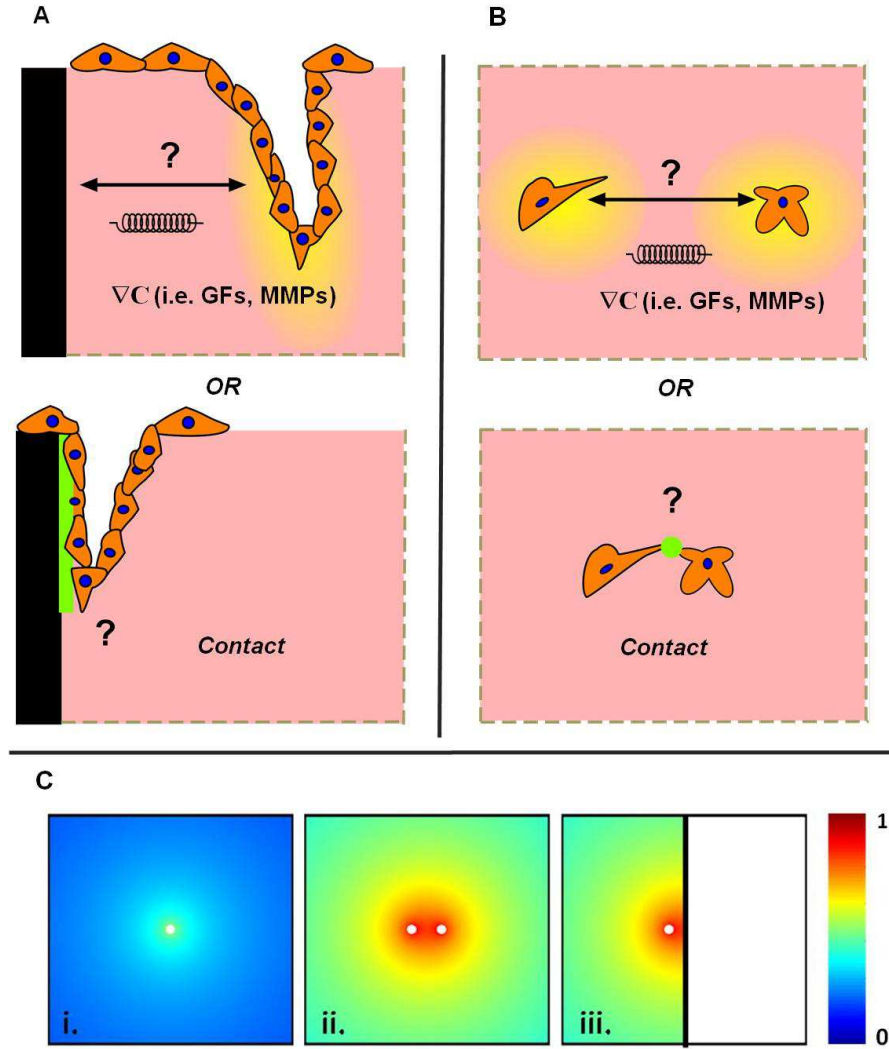


Figure 1. Cellular communication in angiogenesis and vasculogenesis. Multi-cellular structures (A) or individual cells (B) may interact with the microenvironment or neighboring cells. We consider two possible guidance mechanisms that can operate at a distance (top): mechanical interactions (represented by spring) associated with traction forces on a fibrous scaffold, chemical interactions based on the secretion of soluble factors or proteinases, and contact guidance distance interaction. We also consider contact guidance (bottom) in which the membrane of cells make direct contact with other cells or objects. Cells bodies are depicted in orange, nuclei in blue, decaying mechanical or chemical signal in yellow, and contact points in green. (C) Numerical calculations for mass transfer (both steady state) within a permeable, linearly elastic matrix with various boundary conditions studied experimentally in this work. Mass transfer calculations for (i) an isolated cell, (ii) a cell pair separated by X cell-diameters, and (iii) a cell X/2 cell-diameters from a solid, impermeable wall. (Note: at steady state, the distribution of stress and concentration do not depend on the absolute values of elastic moduli and diffusivity).

During vasculogenesis – the self-organization of vascular networks – individual ECs undergo morphogenesis and assemble an interconnected network of capillary-sized vessels. This primitive network serves as the basis for further remodeling via angiogenesis, resulting in a hierarchical vascular network structure [6-8]. In vasculogenesis and angiogenesis, ECs display behaviors such as migration, protrusion activity, and cellular interconnection as part of the process of pattern formation [9-10]. The self-organization involved in vascular morphogenesis presents an interesting platform to study the relative importance of different fundamental mechanisms of interactions between cells [4, 11-15]. In vasculogenesis, individual cells interact locally to establish a network architecture. Meanwhile, sprouting and non-sprouting (e.g., intussusceptions) angiogenesis are network maturation events that change network architecture over distances of many cells. In the context of patterning of vascular networks, studies *in vivo* suggest the importance of guidance cues that act at a distance [16]. For example, Gerhardt, et. al. reported that VEGF gradients control the angiogenic sprouting of ECs that overlay astrocytic networks by guiding filopodial extensions of the tip cell [4]. In addition, contact-dependent signaling via Delta4-Notch1 signaling has also been reported in the vasculature [17].

In vitro, different approaches have been used to characterize how ECs respond to their environmental cues [18]. For instance, Reinhart-King et al., reported that ECs are capable of sensing mechanical forces exerted on substrates by adjacent cells and that this sensing mechanism hinders the migration of cells in pairs promoting the formation of cell-cell contacts [5]. This work suggests that cells in pairs can sense each other at a length scale on the order of cell size, without direct contact. Manoussaki et. al, proposed a mechanical model whereby cellular networks form as a result of traction forces exerted by the cells on the matrix, and suggests that cells do not need to migrate for network formation [12]. In contrast, Serini et al.,

suggest that cell migration is directed by the presence of gradients of secreted VEGF in an autocrine fashion, and that this migration (on the order of a cell diameter) leads to interconnected networks [11]. Yin et. al. reported that EC migration speed and migration patterns were strongly affected by secreted collagen and VEGF when pairwise EC-EC interactions were assessed [19]. Furthermore, Shamloo et al, reported that VEGF gradients can induce EC polarization and chemotactic migration [20].

These *in vitro* models suggest possible mechanisms for network formation of ECs in two-dimensions (2D), with cells seeded on substrates coated with Matrigel™, collagen or fibronectin (Fn)[5, 11, 21]. However, differences between 2D and 3D in changes in cell morphology, adhesion, migration, and cell-cell communication as well as in cell contractility and cell-induced matrix deformations have been demonstrated [22]. An additional physiological consideration is that in 2D systems, ECs typically fail to form luminal spaces (at short experimental times [9]), which is a functionally key event *in vivo* [13] and is relevant for tissue engineering applications. ECs seeded within native Type I collagen recapitulate important features of vasculogenesis *in vivo* such as protein expression (i.e. CD31 [23], VE-cadherin [24] and matrix metalloproteinases (MMPs) [10, 25-26]) and network characteristics such as interconnected open-lumen structures [13, 26]. Importantly, studies of cell-cell communication have often focused on a single mode (i.e. either chemical or mechanical) without consideration of their cross-talk or simultaneous action [3]. For instance, whether an increase in collagen modulus translated into an increased production of VEGF, or vice-versa, has not been assessed. In this sense, the decoupling of fundamental modes of communication has not been achieved in order to identify their relative importance. Based on these considerations, we argue that a complete understanding of the mechanisms of cell communication during vasculogenesis remains outstanding.

As stated above, ECs can serve as a source of both soluble signals [10-11, 25] and of mechanical perturbations on the ECM (**Fig. 1A-B**) [5, 27]; both of these types of signals can mediate interactions between cells that are separated in space, with no direct contact of their membranes. Additionally, ECs communicate during direct contact via membrane-bound receptors [17] (**Fig. 1A-B**). In this work, we studied the migratory, protrusive, and connective activity of ECs during vasculogenesis and sprouting angiogenesis in a permissive [9] 3D collagen environment: ECs were allowed to progress to construct vessels, without external manipulation of signals (i.e. chemical). With this approach, we have attempted to distinguish between the dominant modes of cellular interaction in these two contexts. In particular, we aim to differentiate between non-contact-based (chemical and mechanical) and contact-based interactions during vasculogenesis and angiogenesis *in vitro*. For this purpose, we tracked individual cells within the vicinity of a neighboring cell and the progression of angiogenic sprouts within the vicinity of an impermeable, solid boundary, respectively. Here, an impermeable, solid boundary is a boundary on the ECM which inhibits the passage of diffusible molecules and which with an elastic modulus that is large compared to typical stress generated by a cell. An individual cell or sprout acts as a source of soluble factors or stress (**Fig. 1A-B**), and it will experience a distribution of concentration and stress (**Fig. 1Ci**). A signal generated by two adjacent cells or sprouts will accumulate within the space between them (**Fig. 1Cii**). A solid impermeable boundary acts as a reflective boundary with respect to both soluble factors and stress (**Fig. 1Ciii**); the reflection is equivalent to the accumulation between pairs of cells or sprouts (**Fig. 1Cii**); the reflection presents the cell with an “image” of itself [28]. With these considerations, if these chemical and mechanical factors mediate interactions between cells, then

the gradient generated between a pair of cells or a cell and a boundary could lead to biased behavior.

2-2 MATERIALS AND METHODS

2.2.1. Reagents

We used the following cells, reagents, and materials: Human Umbilical Vein Endothelial Cells (HUVECs), 1x Medium 199 (1x M199), fetal bovine serum (FBS), L-glutamine, penicillin/streptomycin, trypsin/EDTA, HEPES buffered saline solution (Lonza); Endothelial Cell Growth Supplement (ECGS), basic fibroblast growth factor (FGF), phosphate buffered saline (PBS) (Millipore); Vascular endothelial cell growth factor (VEGF) (R&D Systems); Rat tails (Pel-Freez Biologicals, Rogers AR); Phorbol-12-myristate-13-acetate (TPA) (Cell Signaling Technology, Danvers, MA); (poly)ethylenimine (PEI)(MW 750,000; Sigma-Aldrich, St. Louis, Missouri); Heparin solution, Alexa Fluor 488 phalloidin, Alexa Fluor 568 phalloidin, 4',6-diamidino-2-phenylindole (DAPI), 10x Medium 199 (10x M199) (Invitrogen); Acetic Acid (Mallinckrodt Chemicals); L-ascorbic acid (Acros Organics, Atlanta, GA); Bovine Serum Albumin, Triton X-100 (MP Biomedicals, Inc., Solon, OH); (poly)-dimethylsiloxane (PDMS) (Sylgard[®] 184, Dow Corning, Midland, Michigan); Glass discs (Edmund Optics); Glass coverslips (Fisher Scientific); Sodium Hydroxyde (NaOH) (VWR); Biopsy Punches (Miltex by Kai); Glutaraldehyde(Fluka); Formaldehyde (Polysciences, Inc.).

2.2.2 Cell Culture Preparation and Collagen Stock Solution

HUVECs were used in passages 3-9. For expansion and maintenance, cells were cultured in M199 containing 20% FBS, 30 µg/mL of ECGS, 5 Units/mL of heparin solution, 200mM of

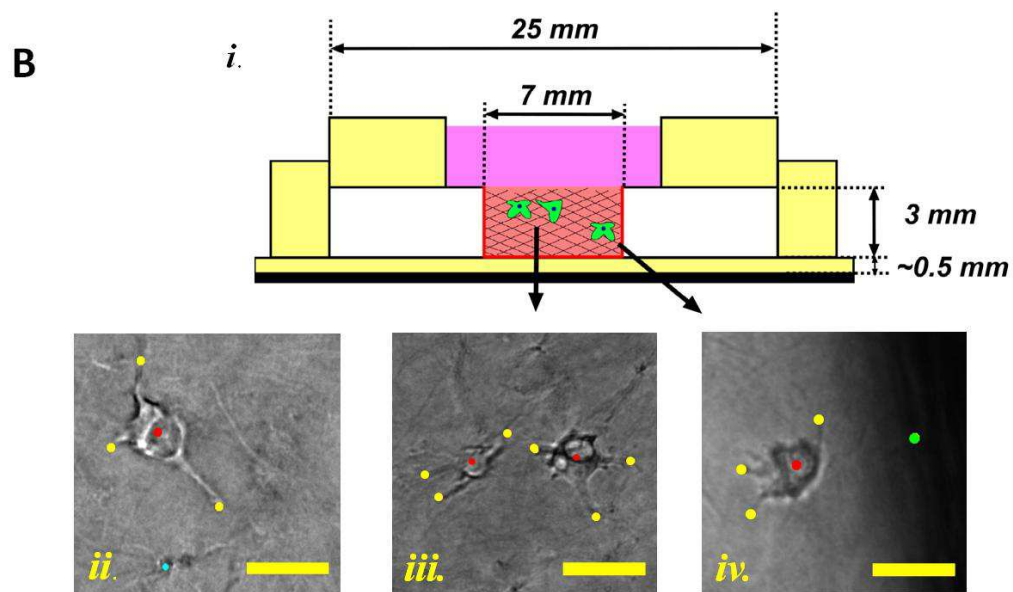
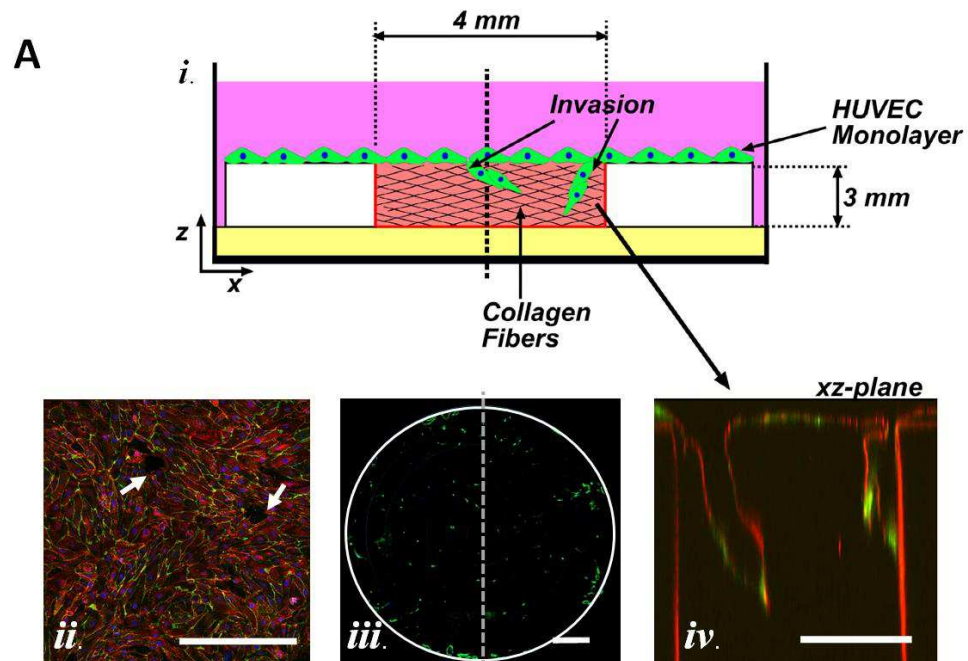
L-glutamine, and 120 U/mL of penicillin/streptomycin. This growth media (GM) was changed every 2-3 days.

Collagen was extracted from rat tails as described previously [29]. Acidic collagen solution was then lyophilized, and the dry mass was determined. The collagen was resuspended in 0.1% v/v acetic acid to 8 mg/mL stock solution.

2.2.3 Invasion Angiogenesis Assays.

To track angiogenesis, we used an experimental system shown in **Fig. 2A**. Glass wells 4 mm in diameter were drilled half-way through the thickness of the disc. PDMS wells 400 μm and 50 μm in diameter were made by casting uncured PDMS onto an SU-8 master with the geometric features. Cleaning, activation, and gel making were performed as described above, with the exception of the addition of cells to the collagen mixture. A HUVEC monolayer was deposited at a surface density of 250 cells/ mm^2 . Cells were allowed to attach for 3 hrs, and then cultured in VM for 7 days with changes of medium every 2 days.

Invasion assay experiments were stopped at different time-points by submerging the samples in 3.7% formaldehyde in PBS for 30 min. Then they were washed 3 times for 10 min in PBS, permeabilized/blocked for 30min with 0.1% Triton X-100/3% BSA in PBS. Gels were then incubated for 1 hr with a 1% BSA solution containing 1:100 dilution of Alexa Fluor 488 phalloidin or Alexa Fluor 568 phalloidin and DAPI 5 μM . A Zeiss 510 or 710 confocal microscope with a water immersion 25 \times objective, was used. Invasions were observed by inverting the invasion experiment wells and imaging slices 0 μm , 10 μm , 20 μm and 50 μm from the monolayer ($z = 0 \mu\text{m}$) (**Fig. 3A**). Fluorescence (**Fig. 2Aii-iii**) and reflectance (**Fig. 3C**, pinhole $\gg 1$) images were obtained.



□ Permeable
or Impermeable Boundary

■ PDMS Seal

■ Cell Media

■ PEI/GA
Adherent Boundaries

■ Glass Cover Slip

Figure 2. Depiction of invasion angiogenesis and vasculogenesis formation assays. (A) Invasion angiogenesis assay. (i) A confluent monolayer of HUVECs (green) invades into the underlying collagen scaffold days after seeding. Collagen is contained in a solid, permeable (2% collagen) or solid, impermeable (glass or PDMS) well with activated surfaces to bind collagen as described in Methods. Each individual well is submerged in vasculogenesis media inside a well plate. (ii) Fluorescence confocal micrographs (25× magnification, zoomed at 0.6; tiled 2×2) of monolayer 1 day after seeding on collagen (green: CD31, red: actin; blue: nuclei). Arrows indicate invasions. Scale bar = 500 μm . (iii) Fluorescence micrographs showing an entire well (2 mm-radius) at a depth, $z = 50 \mu\text{m}$ beneath the monolayer (green: actin). Scale bar = 500 μm . (iv) Fluorescence confocal micrograph (magnification) of vertical cross-section showing lumenized invasions into a collagen-filled well (25 μm -radius) (red: actin; green: nuclei). Note light scattering from or non-specific adsorption on walls of well make these structures red. Scale bar = 25 μm . (B) Vasculogenesis assay. (i) A solid impermeable (glass or PDMS) boundary well sealed with PDMS holds HUVECs seeded into 0.3% collagen and submerged in media. Bright field micrographs (40× objective) of an isolated cell (ii), a cell pair (iii), and a cell next to a wall (iv). Scale bar = 30 μm . Red is center of cell body, yellow dots label the maximum length of a protrusion, green dots label reference point. Color code for illustrations in (A) and (B).

2.2.4. *Luminal density image processing and calculation in 4mm-diameter wells*

The stacks of images at different distances from the monolayer were processed using Matlab® (AppendixD-1). For every image the custom Matlab® code performed a constant size cropping on the images to include only the well cross-sectional view (**Fig. 3Ci**). The cropped images were thresholded ($th = 0.3$) and binarized. The image was then reversed to create positive structures in the void spaces that represent a luminal structure (**Fig. 3Cii**). The reversed image was then segmented concentrically into 4 rings of equal area using the relation,

$$r_i = r_1 \sqrt{i} , \quad (1)$$

where r_1 is the radius of the inner-most ring (in this case a circle) and r_i is the radius of ring i . The equal area rings were defined by the radii calculated using **Eq. 1**. **Figure 3B** depicts “Ring 3” and **Fig. 3Ciii** shows the product of the process steps. The number of solid objects in an image containing solely a particular ring was counted to obtain the luminal density per ring segment. The luminal densities in PDMS wells with diameters of 400 μm and 50 μm were counted manually by identifying structures of luminal invasions (**Fig. 5b, c, e, f, h, i, k, l**).

2.2.5. *Vasculogenesis and Network formation assay*

Cells at 75-95% confluency were washed with HEPES buffered saline solution, removed from culture flasks with trypsin-EDTA, neutralized with GM, centrifuged, and resuspended to a density of 3×10^6 cells/mL. Cell seeded collagen gels were prepared by mixing 10x M199 (1:10; compared to the desired final mixture volume), NaOH 1N (1:50, compared to the required collagen stock volume), 1x M199 (as the last component added to complete final mixture volume), resuspended HUVECs, and stock collagen solution. The working concentration of collagen was 3 mg/mL and the density of HUVECs 1×10^6 cells/mL. **Figure 2B** shows a

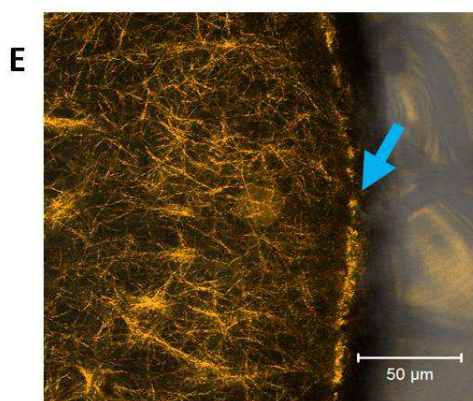
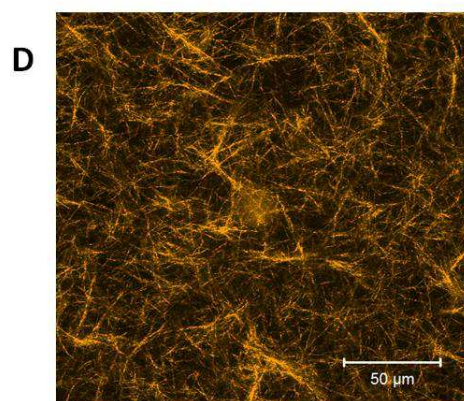
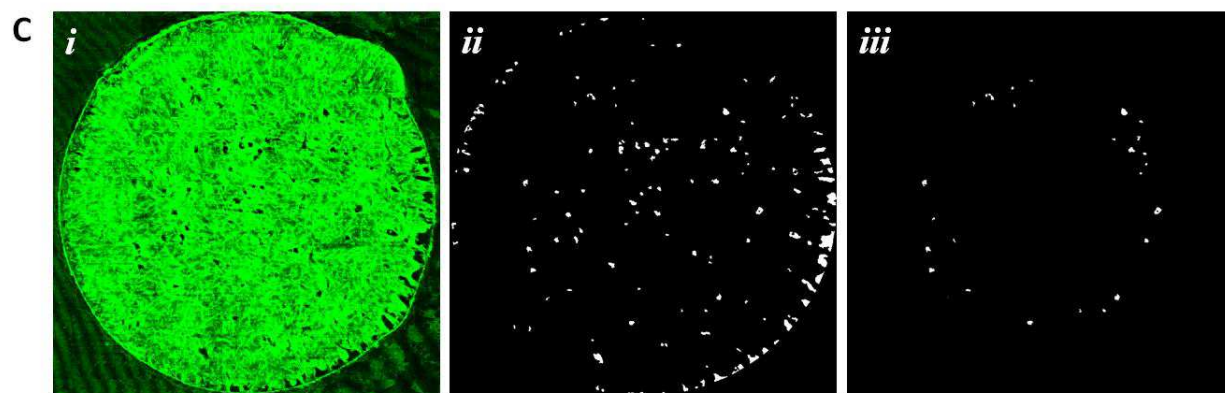
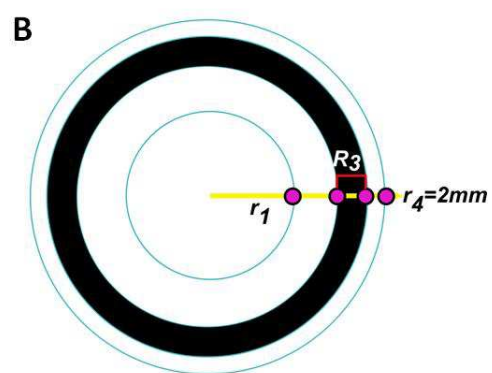
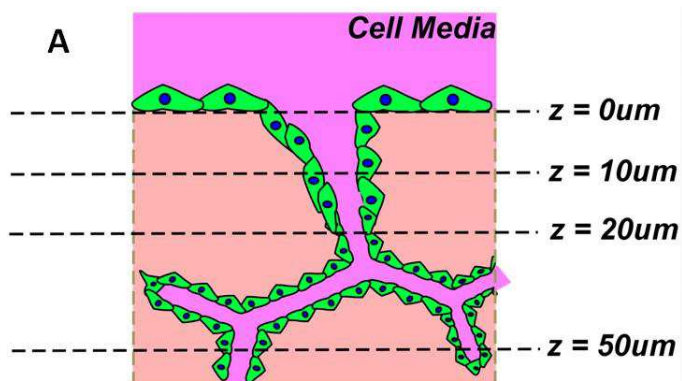


Figure 3. Analysis of invasion angiogenesis assay. (A) Schematic representation of cross-section of invasion from a HUVEC (green with blue nuclei) monolayer into collagen; luminal spaces are depicted in white. Confocal images were taken at $z = 10, 20$, and $50\ \mu\text{m}$ beneath the monolayer. At each depth, a branched invasion is counted as the number of branches. (B) Schematic depiction of the division of the well area into equal areas for analysis. A 4 mm-diameter well was segmented into regions (R_i) of area = $3.14\ \text{mm}^2$ (boundaries at radii of 1 mm, 1.41 mm, 1.73 mm, and 2 mm). (C) Image analysis of the spatial distribution of luminal spaces. (i) Confocal reflectance image of collagen fibers (green) within a 4 mm-diameter glass well (pin-hole much greater 1; $z = 20\ \mu\text{m}$ shown). (ii) Thresholded, binarized, and inverted reflectance image (from (i)) is used to identify individual lumens. (iii) The pre-processed image (from (ii)) was segmented into concentric rings of equal areas as depicted in (B). The lumens identified within third ring (shaded in (B)) are shown. (D, E) The location inside a well does not affect collagen fiber architecture. Reflectance microscopy images of collagen fibers are shown in orange (false color). (D-E) Reflectance confocal images ($25\times$ objective) of collagen fibers in the center of the well (D) and at wall (E). Blue arrow in (E) indicates glass boundary. Scale bar = $50\ \mu\text{m}$.

schematic of the network formation assays. Holes, 4mm in diameter were drilled on a 3 mm thick glass disc (20 mm diameter). A PDMS thin layer was deposited on a cleaned cover slip. The glass disc was pressed manually onto partially cured PDMS (25 min at 60°C) on the cover slip and allowed to cure completely (2 hrs at 60°C). Prefabricated slabs of PDMS (with 10 mm diameter holes punched) to serve as reservoirs (5-6 mm high) were pressed onto the outer surfaces of the glass disc with uncured PDMS solution. PDMS was then cured (2 hrs at 60°C). The well systems were then cleaned and the glass component was treated to covalently bind the collagen to the well boundaries to prevent cell-mediated gel contraction [30]. Cell seeded collagen solutions were placed in dried, pre-chilled (4°C) wells and allowed to polymerize in the wells inside an incubator at 37° C and 5% CO₂ for 20 min. A volume of 350 µL of GM containing 50 µg/mL L-ascorbic acid, 50 ng/mL TPA, 40 ng/mL VEGF, and 40 ng/mL bFGF was added to each well. We call this formulation Vasculogenesis Media (VM). Wells with VM were placed in the incubator for 72 hrs.

Network formation experiments were stopped at different time-points by submerging the samples in 3.7% formaldehyde in PBS for 30 min. Then they were washed 3 times for 10 min in PBS, permeabilized/blocked for 30min with 0.1% Triton X-100/3% BSA in PBS. Gels were then incubated for 1 hr with a 1% BSA solution containing 1:100 dilution of Alexa Fluor 488 phalloidin or Alexa Fluor 568 phalloidin and DAPI 5µM. We used a Zeiss 510 or 710 with a water immersion 25× objective, and focusing at a distance of 200 µm into the gel to image stained cells (**Fig. 8A-B**).

2.2.6. Individual Cells Protrusion Tracking Assay

To track individual HUVECs (**Fig. 2Bii-iv**) we used the same device and formulations described above, with a working cell density of $1-1.5 \times 10^5$ cells/mL. Fully loaded (collagen

seeded with cells) devices were placed in an incubator (Weather Station) mounted on an Olympus IX81 microscope stage to maintain a temperature of 37°C and imaged with a 40× objective. A smaller, custom-made Plexiglas chamber, lined with wet paper towels, was used to maintain high humidity around the cell culture. A 100% CO₂ stream mixed with air produced a 5% CO₂ stream that was bubbled through a column of water and flowed directly into the small custom chamber and over the culture. An automated stage control (SlideBookTM) was used to record the *xyz*-coordinates for the locations of interest. Multiple cells (or cell pairs, **Fig.2Bii**) were analyzed from a single well.

Protrusive activity of cells in collagen was tracked by capturing images of individual cells every 10min, using a Hamamatsu ORCA-ER camera mounted on an Olympus IX81 microscope with a 40× LWD objective. A set of 3 images, spaced by 25 µm in the *z*-direction, were collected at each position at each time point resulting in a *z*-stack (75 µm-deep) centered around the original center of every cell for every refocused plane. Time lapse experiments were taken for 24 hrs with manual re-focusing of the stage every 2 hrs. Cells or cell pairs found at distances greater than 250 µm from neighboring cells and the bottom boundary were identified and their location recorded. Such individual cells or pairs of cells were considered to be isolated with respect to the interactions considered. Single isolated cells (**Fig. 2Bii**), cell pairs of different initial separations (**Fig.2Biii**), and single cells of different separation from a vertical boundary (**Fig. 2Biv**) were tracked.

2.2.7. Analysis of Protrusive Activity and Cell Migratory Activity

The in-focus plane of every time point was selected to generate a movie of cell behavior over the 24hr period. The *xy*-coordinates of the cell centers (red dots in **Fig. 2B ii-iv**), a reference

location or boundary (blue dots in **Fig. 2B ii-iv**) to account for whole image displacement, and the boundary (when applicable) were identified manually using Image J software (**Fig. 2Bii-vi, Fig. 4A**). The xy -coordinates of the end of each protrusion (yellow dots in **Fig. 2B ii-iv**) at its maximum length were identified manually (**Fig. 4E**). The average protrusion frequency and length were calculated. The migratory activity of individual cells was quantified using the displacement from origin as follows:

$$D(t_k) = \left\langle \sqrt{[x(t_k) - x(t_1)]_i^2 + [y(t_k) - y(t_1)]_i^2} \right\rangle \quad (2)$$

where i is an individual cell, k represents a time step, $x(t_k)$ and $y(t_k)$ coordinates of the cell center at every time point, and $x(t_1)$ and $y(t_1)$ are the cell center coordinates of the original location. The brackets represent that the average was taken over all i cells.

2.2.8. Coordinate Transformation

We were interested in the potential directionality of the set of protrusions from one cell towards a neighboring cell or a boundary. To account for the displacement of both cell centers (or the direction of protrusions towards a boundary) we transformed the protrusion coordinates using **Equation 3**, from the frame of reference of the collected image (**Fig. 4B**, black) to the frame of reference that contains the two cells at every time-point (**Fig. 4B**, blue):

$$\begin{pmatrix} x''_p \\ y''_p \end{pmatrix} = \begin{pmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{pmatrix} \begin{pmatrix} x'_p \\ y'_p \end{pmatrix}, \quad (3)$$

where, (x'_p, y'_p) are the coordinates in the frame of reference the center of mass for an individual cell, (x''_p, y''_p) are the coordinates in the frame of reference where the x-coordinate for both cells are aligned, and ϕ is the angle between two centers of mass. The transformation only

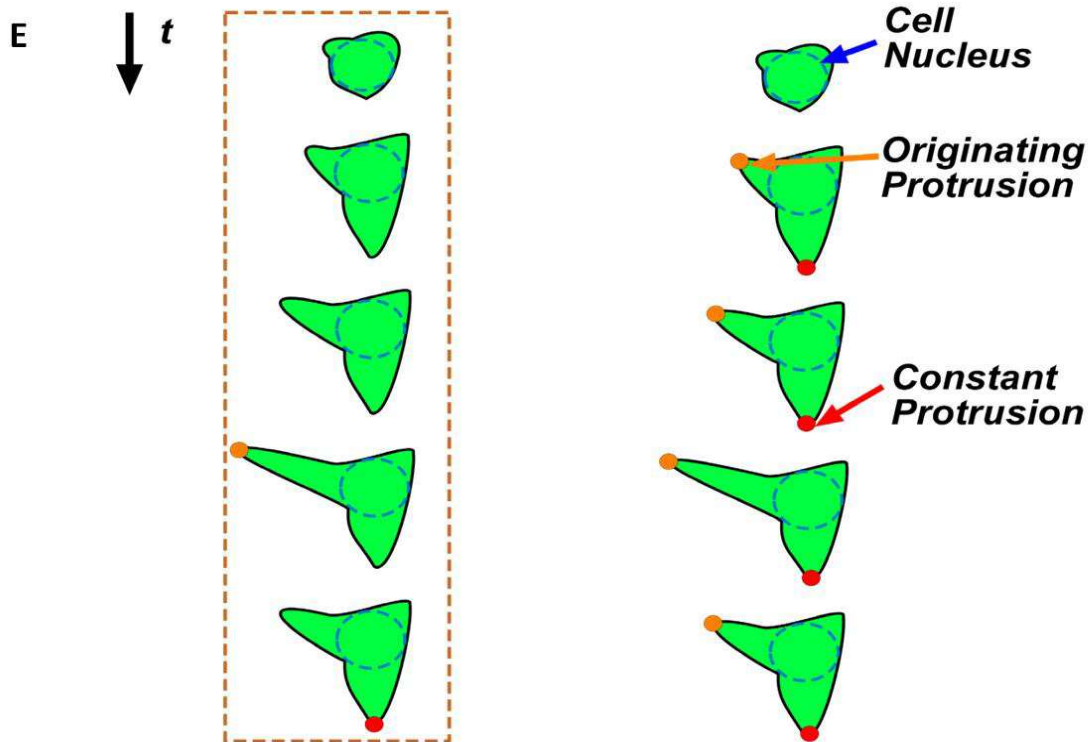
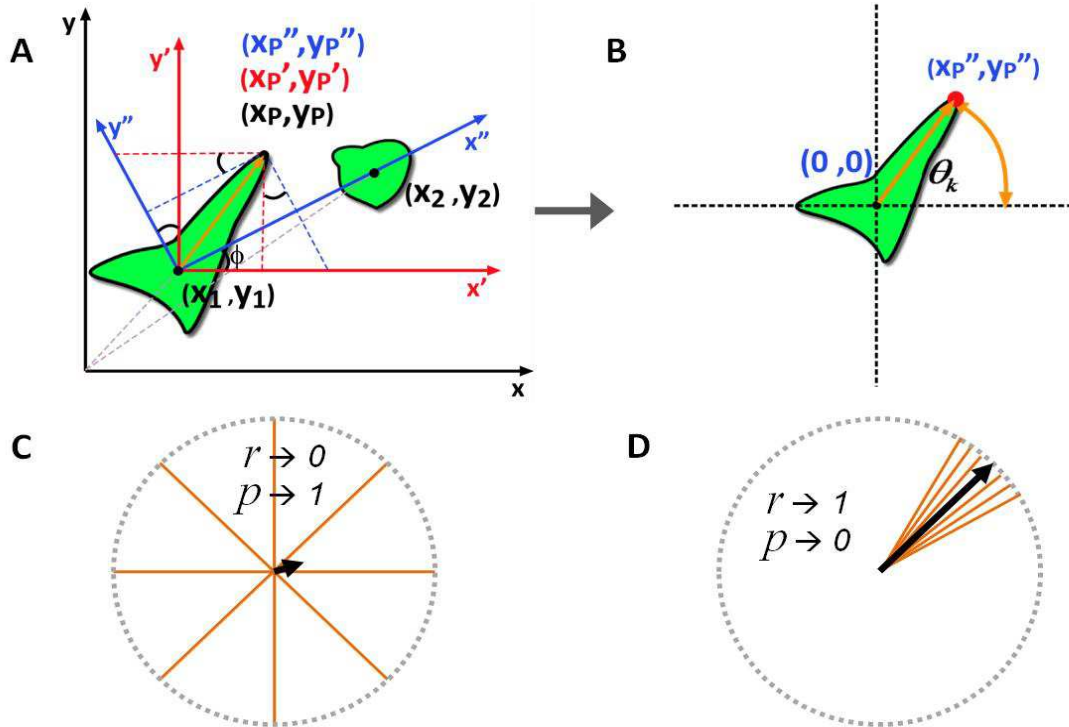


Figure 4. Schematic representation of analysis of orientations of protrusions. (A) For cell pairs and cell-boundary experiments, we use two reference frames: (x, y) with its origin on one of the cell's nucleus and aligned with the borders of the image and (x', y') rotated by an angle ϕ such that the x' -axis passes through the nucleus of the neighboring cell (cell pairs) or through a fixed point on the boundary. The transformation in Eq. 1 transforms between these two frames. (B) The angle θ_k of the k^{th} protrusion is measured with respect to the x' -axis, as indicated by Eq. 2. (C-D) Schematic representations of two possible angular distributions of protrusions transformed with Eq. 3 onto a unit circle: uniform (C) for which the resultant length, r tends to 0 and the p -value for the circular statistic tends to 1; concentrated or biased (D) for which the resultant length, r tends to 1 and the p -value for the circular statistic tends to 0 [31]. (E) Selecting a protrusion. A set of protrusion were extended towards the matrix and remained extended at its original location in the matrix ("constant protrusion", red dot) while others extended and retracted (orange dot). The center of mass for the cell structure was selected to be the center of the cell body.

depends on the angle between two cell centers (or cell center and boundary), namely ϕ , and the xy -coordinates of the protrusions with respect to their respective cells (x'_p, y'_p) (shown in blue, **Fig. 4B**). The transformation sets the angle between the neighboring cell or the boundary and the reference cell to zero degrees for every time-point. These calculations were carried out by a custom Matlab® code (Appendix D-2 through D-4) that includes the Circular Statistics calculations (next section).

2.2.9. Circular Statistics

Statistical methods for circular data [32] were employed to characterize the angular distribution of protrusions within the frame of reference, (x''_p, y''_p) defined by a cell pair or a cell and the boundary (**Fig. 4A-B**). The Rayleigh test evaluates whether there is directedness in a sample and the V-test assesses if the directionality of a sample is towards a predicted direction [32]. The resultant length (r) is a parameter of concentration in the Rayleigh test and is calculated as,

$$r_i = \sqrt{\left\langle x''_P \right\rangle_i^2 + \left\langle y''_P \right\rangle_i^2} \quad \bar{\theta}_i = \tan^{-1} \left(\frac{\left\langle y''_{P,k} \right\rangle}{\left\langle x''_{P,k} \right\rangle} \right), \quad (4)$$

where i is an individual cell, $\left\langle x''_{P,k} \right\rangle$ is the average of unit x -coordinate for k protrusions. The closer the value of r to 1 is, the higher the directionality (**Fig. 4C-D**). The p -value is the probability of observing an event at least as extreme as the one observed, when the null hypothesis (that the data set is uniformly distributed around a circle, as in **Fig.4C**) is true. The null hypothesis of a statistical hypothesis test is rejected when the p -value is less than the pre-

assigned significance level α (0.05 for our statistical tests). A p -value for the Rayleigh test depends on the number of protrusions (N), the resultant length (r), and can be estimated with,

$$p_i = \exp \left[\sqrt{1 + 4N_i + 4 \left(N_i^2 - (N_i r_i)^2 \right)} - (1 + 2N_i) \right] \quad (5)$$

The protrusions of individual cells were subjected to the Rayleigh test and if the null hypothesis was rejected (null hypothesis: uniformity of distribution), we used the V-test to ask if the protrusions were directed towards a neighboring cell (angle equal to 0). The test statistic (u) for the V-test takes into account the parameters from the Rayleigh test (**Eq.4**) and N , the number of samples. It can be calculated using **Equation 6**, where ϕ is the predicted direction, and the decision values are tabulated [32].

$$u = r\sqrt{2N} \cos(\bar{\theta}_i - \phi) \quad (6)$$

2-3 RESULTS

2.3.1. Invasion Angiogenesis

Invasion angiogenesis represents a multicellular process [4, 17, 33] (**Fig. 1B**). An angiogenic invasion is guided by an exploratory tip cell that also regulates the identity of the neighboring cells in the invasion [4]. Here we describe the interaction of these multicellular processes with the microenvironment. We address the interaction of individual cells during vasculogenesis in section 3.2.

2.3.1.1. Qualitative features of invasions in wells

We filled wells with 0.3% collagen then we seeded the collagen surface with a confluent monolayer of HUVECs (**Fig. 2Aii**). Wells were made of glass or PDMS, which represent solid, impermeable boundaries. In a previous study, Yannas, et. al., reported that the regenerative capacity of ECM grafted on standard, full-thickness skin wounds in the adult guinea depended on ECM pore size [34]. In addition, we previously found that the size of pores in microstructured tissue templates (MTTs) impacted the guidance of cellular invasion [35]. Zheng, et. al., report that when such MTTs were implanted into mice, the smaller pore diameters (constant well depth) induce higher cell infiltration into the wells [35]. To assess any potential differences in our sprouting angiogenesis assay due to geometrical constraints, we used wells of multiple diameters (50 μm , 400 μm , and 4 mm). In wells of each diameter, cultures were assessed at different depths from the monolayer: (1) *Surface*: At $z = 0 \mu\text{m}$, deformations on the order of many cells caused the monolayer to appear patchy as seen by dark areas within a well (**Fig. 5a**). In 400 μm -diameter and in 50 μm -diameter wells these deformations appear to be covering an entire well area (**Fig. 5b-c**). For 50 μm -diameter wells, 3-4 cells are seen attached the wall of the well (**Fig. 5c,f,i,l**, green nuclei). (2) *Bulk*: Beyond $z = 10 \mu\text{m}$, individual invasions are evident in the form of discrete dark regions inside the wells (**Fig. 5, d-k**, examples are labeled with white arrowheads). Angiogenic invasions were generally found associated with or in the vicinity of a boundary, and more sparsely in the interior of the well (**Fig 5, h-i**). In 50 μm -diameter wells, a single luminal space suggests that only one invasion occurred (**Fig. 5i**, star).

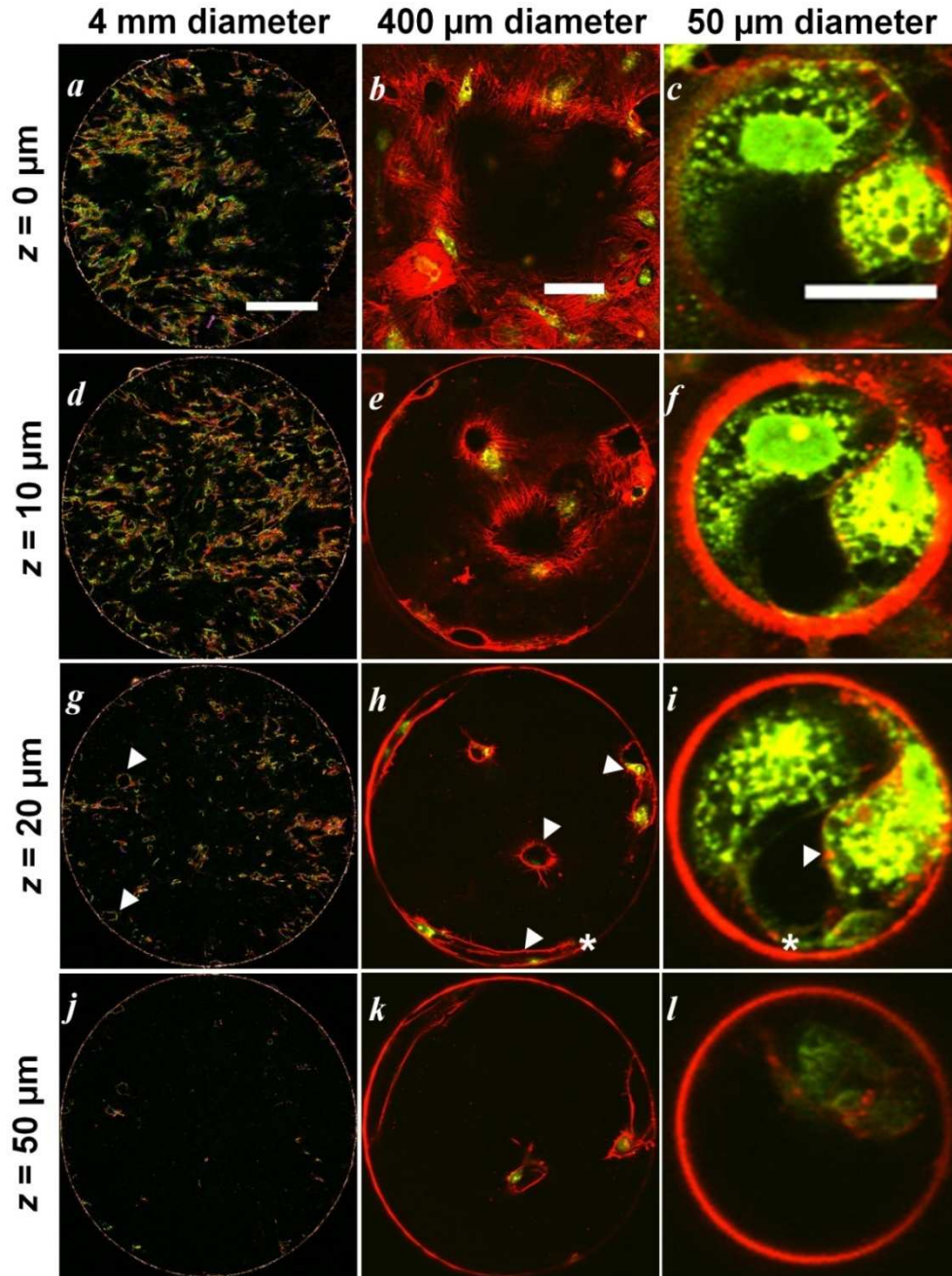


Figure 5. Invasion cultures in wells of different diameters. Confocal fluorescence micrographs of invasion assays using well-diameters of 4 mm (left column), 400 μm (center column), and 50 μm (right column) at different depths below the monolayer (0 μm, 10 μm, 20 μm, 50 μm). For 4 mm wells (left column), red is actin, green is CD31, and blue is DAPI staining nuclei; scale bar = 1 mm. In the columns for images of 400 μm (PDMS) and 50 μm-diameter wells (PDMS), green is YOYO-1 staining nuclei and red is actin. In the row for z = 20 μm, white arrowheads represent luminal structures found 7 days after culture (not all luminal structures are labeled). White stars indicate cell membranes closely associated with the wall in several luminal structures.

2.3.1.2. Variations in global luminal density with well diameter

The qualitative examination of invasions into wells of different diameter suggests that the number of invasions varied with the diameter of the well (**Fig. 5**). **Figure 6** shows an increase in global luminal density per well with decreasing well diameter. Among well diameters, we found the differences in mean global luminal density per well to be significant. The high density seen for 50 μm -diameter wells corresponds to a single invasion in every well. We also note that, for each size of well, the global luminal density did not vary significantly with depths (**Fig. 6**). The significant dependence of luminal density on well diameter suggests that the proximity to a boundary affects the emergence of invasions. We dissected the 4 mm and the 400 μm -diameter wells in the next section to address a potential preference of invasions near the boundary.

2.3.1.3. Boundary effect on HUVEC multicellular angiogenic sprouts

Based on the diameter-dependence observed in **Fig. 6**, we hypothesized that invading sprouts were attracted to the well boundary and as a result, more invasions would be found at the boundary compared to the inner regions of the well. In addition, we wanted to analyze these observations with respect to our original hypothesis that cells emit signals that act at a distance (**Fig. 1A, C**). To test these hypotheses, we analyzed the density of lumens in annular segments of equal areas in the 4 mm and 400 μm -diameter wells (**Fig. 3B** and Methods). **Figure 7** shows the segmentation of 4 mm (**A**) and of 400 μm (**B**) wells with the associated luminal density in each radial segment. This segmentation was performed at depths of 20 μm , and 50 μm from the monolayer. We performed one-way ANOVA tests using 95% confidence intervals on the grouped data sets for luminal density in a ring as a function of radial positions for both depths (20 μm and 50 μm). Luminal densities showed significant differences at different radii only at a

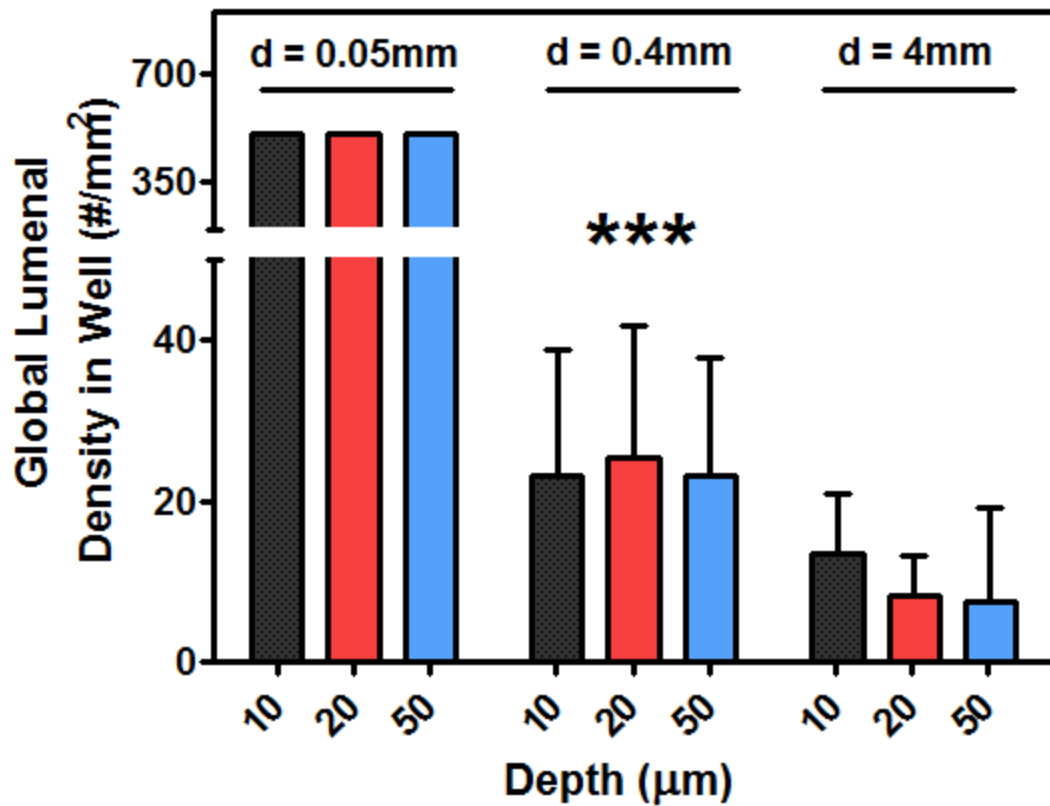


Figure 6. Wall effect on luminal density. Global luminal density per well were plotted for different depths in a well, and different well diameters. Means of global luminal density were found to be significantly different across different well diameters (***) $p < 0.001$ and not among different well depths. All well diameters are significantly different from each other in their luminal density per well. Exactly one invasion per well was found for all 0.05 mm wells analyzed. $N = 3$ for $d = 0.05$ mm; $N = 10$ for $d = 0.4$ mm; $N = 10$ for $d = 4$ mm. Error bars: SD.

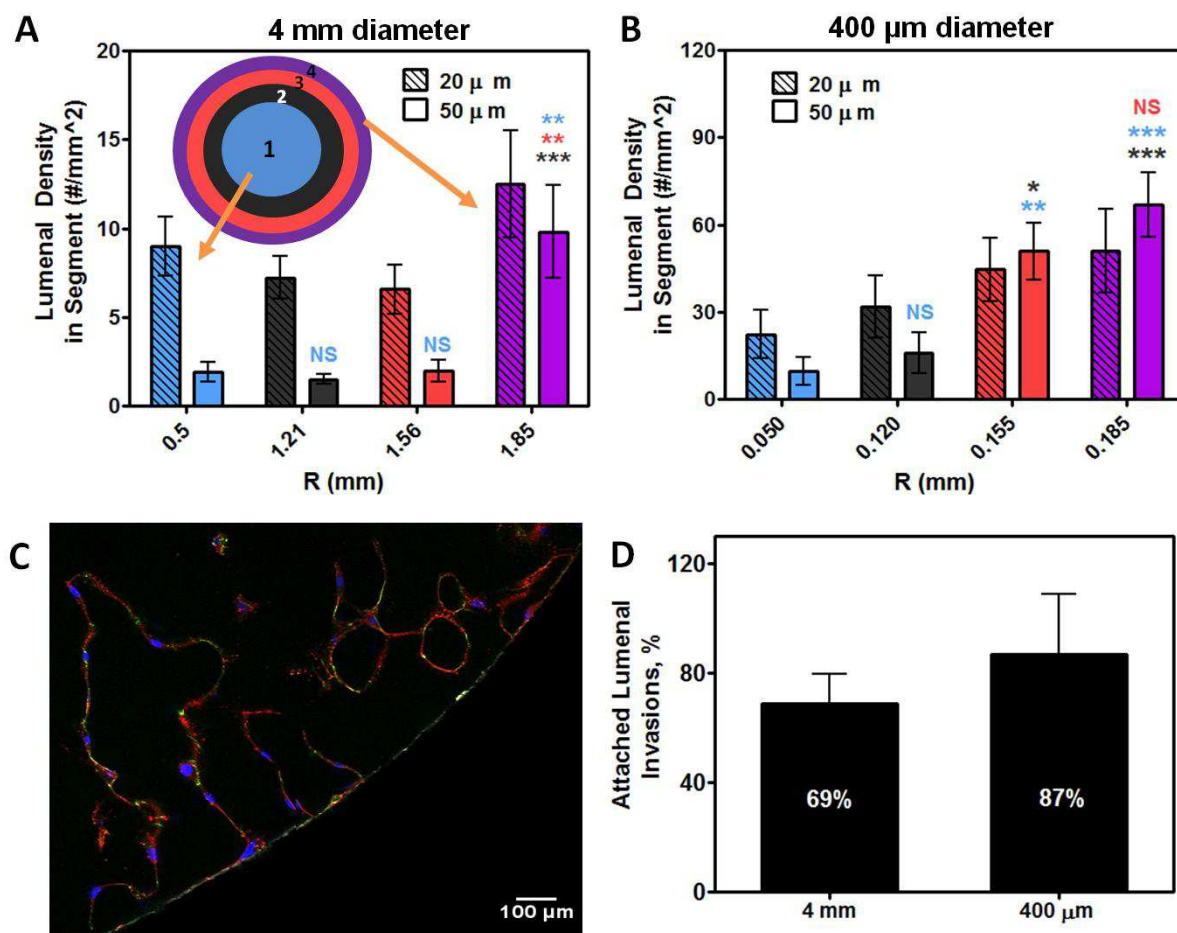


Figure 7. Analysis of density of invasions in concentric annuli of equal areas. (A, B) Luminal invasion density was calculated for individual annuli. One-way ANOVA tests using 95% confidence intervals were performed on the grouped data sets for luminal density in a ring as a function of radial positions; this test was followed by a Tukey's Multiple Comparisons Test using a 95% CI. Averages of luminal densities were found to be significantly different spatially within a well only at a depth of 50 μ m (solid bars). This effect was found for wells with diameters of 4 mm (A), and 400 μ m (B). **(A)** 4 mm-diameter wells. Significant difference found only for the luminal density of the outer most ring (purple). The resulting radii for 4mm-diameter wells were 1mm, 1.41 mm, 1.73 mm, and 2 mm. **(B)** 400 μ m-diameter wells. Significant differences was found among the two inner most and the two outer most rings. No significant difference was found among the outer most rings (red and purple). The resulting radii for 400 μ m-diameter wells were 100 μ m, 141 μ m, 173 μ m, and 200 μ m. Matching colors are used to represent the statistical analysis results of specific comparisons. Error bars: SEM. **(C)** Luminal invasions near boundary are found physically attached to the boundary of the well; membrane and nuclei are seen directly attached at boundary. Green is CD31, red is Actin, blue is DAPI staining nuclei. **(D)** Fraction of luminal invasions attached to the boundary of the well. Student's t-test using 95% confidence interval was performed for the comparison of the average values for 4 mm- and 400 μ m-diameter wells. Error bars: SD. (A,-C) NS: no significance, *: $p < 0.05$, **: $p < 0.01$, ***: $p < 0.001$; N = 10.

depth of 50 μm (solid bars). This effect was found for wells of both diameters. The presence of undulations on the order of a single cell ($\sim 25\ \mu\text{m}$; **Fig. C1** Appendix C) mixed with the distortion of the monolayer due to the origination of invasions (**Fig. 5a,d**) may explain the absence of significant radial variation in density in **Fig. 7A-B** at 20 μm from the monolayer (crossed bars).

In order to identify the origin of the differences for luminal densities at different radial positions in 4 mm-diameter (**Fig. 7A**) and 400 μm -diameter (**Fig. 7B**) wells at a depth of 50 μm , we performed a Tukey's Multiple Comparisons Test using a 95% CI post ANOVA. **Figure 7A** (solid bars) shows that, for 4 mm-diameter wells, no significant differences were found among means of the inner-most radial positions (rings 1-3). However, in all of the inner-most radial positions, we observe significantly lower density relative to the wall-associated data set (outermost annulus, or ring 4). We note that the preponderance of the lumenized invasions in this outer region is due to invasions that were directly attached to the wall of the wells, as seen qualitatively in **Fig. 3C**, **Fig. 5**, and **Fig. 7C**. Analysis of luminal invasions in wells with permeable boundaries (collagen at 2% concentration) exhibit the same qualitative behavior (**Fig. C2** Appendix C); a lack of replicates of these experiments meant that we were unable to perform statistical analysis. **Figure 7B** shows a similar analysis for 400 μm -diameter wells; no significant differences were found among means of the two inner-most (ring 1 and 2) or among the two outer-most radial positions (3 and 4), but the luminal density in the outermost regions (rings 3 and 4) were significantly larger than in the inner ones.

In wells of both 4 mm and 400 μm , the outermost ring shows significantly higher luminal density. The size of the outermost annulus in 4 mm wells is 270 μm . The average width of an invasion was 50 μm (measured for 57 randomly selected invasions found in both 400 μm and 4 mm-diameter wells at 50 μm depth). The size of an invasion is smaller than the size of the

4 mm-diameter outermost annulus (270 μm). The total size of the outermost annulus in 400 μm wells was 59 μm , comparable to the average invasion size. Thus, in the 400 μm wells, the invasions in the outermost were likely to have been in direct contact with the wall. We calculated the fraction of luminal invasions that were attached to the boundary at the outermost annulus for both 4 mm- and 400 μm -diameter wells. **Figure 7D** shows that the preponderance of invasions near the boundary regions was due invasions in contact with the boundary: more than ~70% of the luminal invasions were touching the wall. Furthermore, the outermost ring for wells 400 μm in diameter contained significantly more attached invasions than the outermost ring for wells 4 mm in diameter; 69% and 87%, respectively.

2.3.2. Vasculogenesis

Preceding angiogenesis, blood vessels form through the process of vasculogenesis from individual ECs that have differentiated from their angioblastic lineage [6]. The local behavior of ECs dictates the global morphology of the emerging network of vessels. Here we describe the interaction between individual ECs that leads to the emergence of a network of vessels. We used a similar approach as when studying angiogenesis in the preceding section: individual cells were presented with a boundary or another cell with varying separation and looked for presence or absence of biased behavior based on these microenvironmental conditions.

2.3.2.1 Separation threshold for connectivity

We observed that cells in 3D collagen environments are able to establish connections with one another to form a network of lumenized vessels (**Fig. 8A-B**). We hypothesized that the

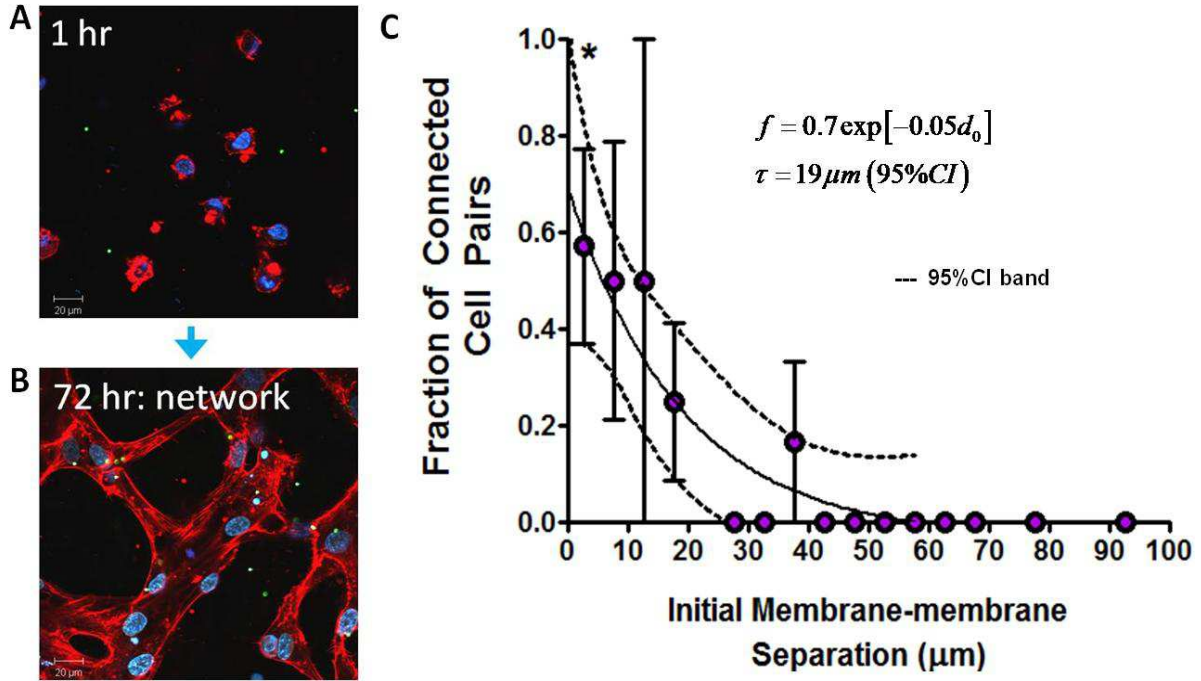


Figure 8. Vasculogenesis in vitro. (A, B) Confocal fluorescent images of fixed and stained HUVECs with AlexaFluor 568 Phalloidin (red = actin) and DAPI (blue = nuclei). (A) The cells are uniformly seeded (at 1×10^6 cells/ml) in the 3D collagen matrix. (B) After 72 hrs of culture, the endothelial cells form a globally connected vascular network. Scale bars = 20 μm . (C) The probability of two cells establishing a connection in a low-seeding cell density 24 hrs ($1-1.5 \times 10^5$ cells/mL) versus initial membrane to membrane separation. We defined a connection as the permanent fusion of vacuolar spaces from two individual cells within the 24 hrs experimental timeframe (from seeding). Cell pairs were tracked every 10 min. Initial cell separations were categorized into 5 μm bins and all cell pairs were sorted into these bins with a value of 1 (connected) or 0 (not connected). A t-test performed on the individual bins assigns significant disparity from a null value. Dashed lines represent the 95% certainty band for the curve fitting. Error bars: SEM (*: $p < 0.05$).

capacity of two individual cells to establish a connection depended on their initial membrane-membrane separation (defined as closest distance between the membranes of a pair of cells that were separated from all other cells by at least 250 μm). A connection was defined as the permanent fusion of vacuolar spaces from two individual cells within the 24 hrs experimental timeframe. Initial cell separations were categorized into 5 μm bins and all cell pairs were sorted into these bins with a value of 1 (connected) or 0 (not connected). When the calculated values for the fraction of connected cell pairs were plotted versus initial separation, the probability of two cells establishing a connection in a low-seeding cell density ($1 - 1.5 \times 10^5$ cells/mL) in a period of 24 hrs decayed with initial membrane to membrane separation (**Fig. 8C**). Non-linear regression of the results to a decaying exponential using a 95% CI validated an exponential decay model with a decay constant of 20 μm . Furthermore, a t-test using a 95% CI performed on the individual bins assigned significant difference from a null value for the fraction of connected cells in the bin for membrane to membrane initial separations of 0 – 5 μm (**Fig. 8C**, star). These observations suggest that individual ECs require the ability to establish physical contact in order to initiate connection and subsequently fuse their vacuoles.

2.3.2.2 Migratory activity of HUVECs in 3D collagen matrices is minimal

Cells were homogeneously embedded in collagen, yet they organized into a connected network of luminal spaces. For two cells to establish a connection within 24 hrs, they needed to start within ~40 μm of membrane to membrane separation (**Fig. 8C**). The migration of individual cells is a suitable explanation to the resulting physical proximity that led to connectivity (**Fig. 8**). To quantify the movement of cells in the bulk collagen, we tracked the centers of mass for individual HUVECs in 3D environments found in isolation or in pairs. We considered an individual cell to be isolated if no neighboring cell or boundary could be found within a distance

of 250 μm in all directions; we considered two cells to be a pair if they were within 40 μm of one another and separated from all other cells and boundaries by more than 250 μm . In **Fig. 9A-B**, we plotted representative collection xy -coordinates for the centers of mass of individual cells through 24 hrs of culture. The trajectories of an isolated cell (**Fig. 9A**) were similar to the trajectories of cell pairs (**Fig. 9B**): cells took small steps that resembled a random walk. To quantify the migration of individual HUVECs we determined their root mean squared displacement from original position (**Eq. 2**) for a period of 24 hrs after media introduction to the culture. **Figure 9C** shows that isolated cells remained on average within 10 μm from their original location (less than the characteristic cell diameter of 30 μm) throughout the 24 hrs experimental timeframe (blue). For all cells, we noted that an individual cell makes displaces from its origin at early times (~ 2 hrs), and then becomes fixed around a specific location (**Fig. 9B**, blue). This fixation time may be related to the time it takes an individual cell to engage with the matrix. In addition, we qualitatively examined and compared the migratory activity of individual HUVECs embedded in collagen matrices to the migratory activity of HUVECs on a flat surface. We observed a dramatic reduction in motility within the 3D environments (**Fig. C4 Appendix C**): a cell fully embedded in collagen showed no significant displacement from its original position (**Fig. 9C**), while cells on a surface moved with large jumps (approximately 50 μm within the first 2 hrs). This dramatic reduction in motility in 3D is consistent with the observation that the 3D fibrous environment presents a physical hindrance for cells to move freely [36].

To assess the potential effect of distant signals (chemical or mechanical) from a neighboring cell on individual cell migratory activity, an analogous characterization on cell pairs was performed. **Figure 9D** shows that the presence of a neighboring cell did not alter the

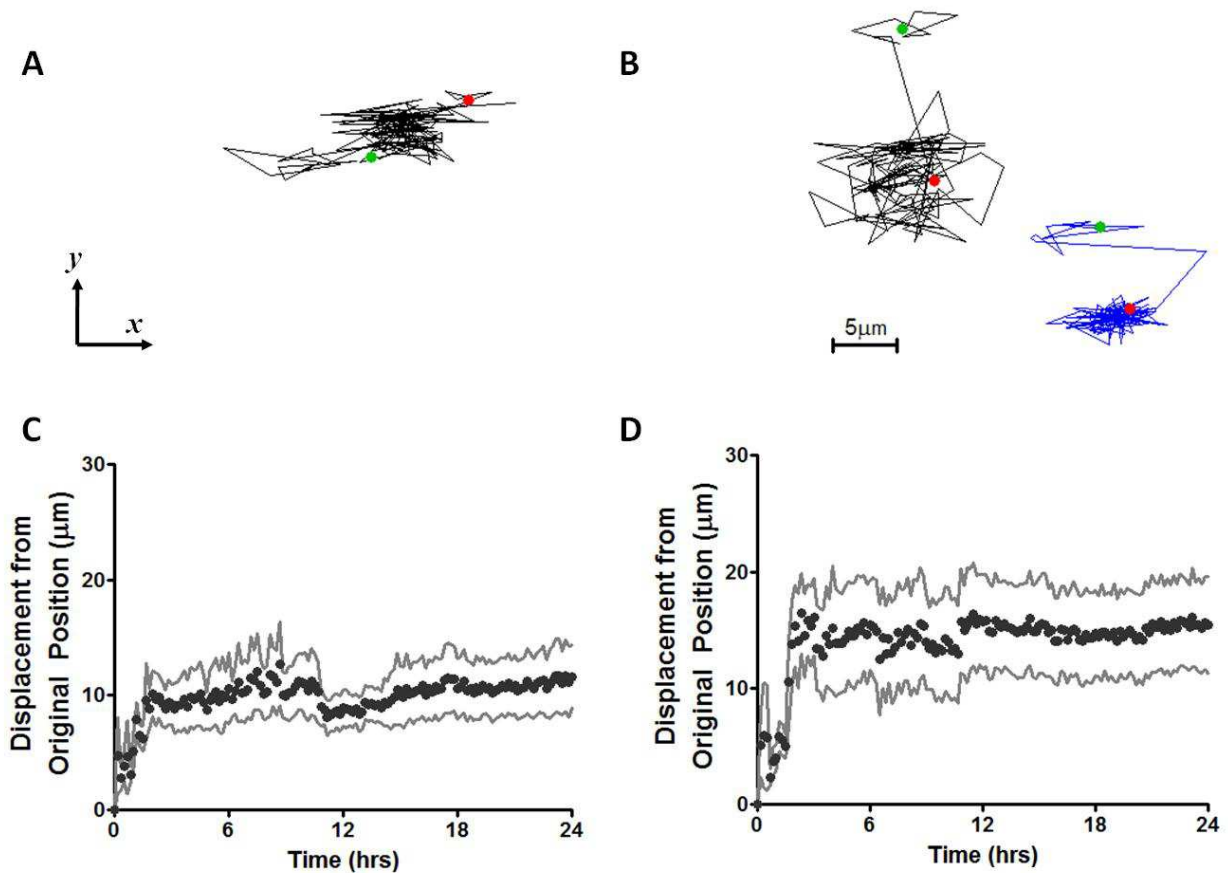


Figure 9. Migration of HUVECs in 0.3% collagen matrices of vasculogenesis. HUVEC migration in 3D collagen scaffolds was tracked for 24 hrs and was characterized using the displacement from original location. (A) xy-locations of cell center of mass for an isolated cell in a 24 hrs period; the time interval between positions is 10 min and the straight lines connect points of subsequent snapshots. Green and red dots indicate original and final location, respectively. (B) xy-locations of cell center of mass for a cell pair (< 40 μm) in a 24 hrs period; black lines represent left cell, blue lines represent right cell. (C-D) The displacement from original position is an ensemble average over all cells for the instantaneous displacement from original location. Displacement from original location for isolated cells (C) and for cells within 40 μm (D). Number of cells within 40 μm = 20; cells greater than 40 μm = 28. Solid lines above and below data points represent 95% CI.

migratory behavior of individual cells. These observations indicate that neighboring cells do not induce changes in motility at this cell-scale and suggests that individual cells use a mechanism other than migration to form vasculogenic connections in 3D collagenous environments. We highlight that these observations are distinct from those in experiments in 2D where individual ECs were responsive to distant signals by altering their migratory activity [5, 11].

2.3.2.3. Individual cell protrusions do not show bias towards a neighboring cell or a boundary

HUVECs in 3D collagen matrices mimic the formation of the embryonic vascular network [13]. One striking behavior of HUVECs during vasculogenesis is their high protrusive activity (vasculogenic sprouts) into the ECM; for instance, 20 cells in pairs generated 920 protrusions (~46 protrusions/cell) in an experimental timeframe of 24 hrs (**Table 1**). Protrusions from one cell can contact neighboring protrusions and connect the individual cells [37]. Accounting for the lack of significant mobility of individual cells in 3D collagen matrices during early time-points of vascular morphogenesis (**Fig.9**) and that VEGF regulates the protrusive activity of ECs [38], we addressed the possibility that the cell's protrusive activity could be a biased event that aided in the formation of connections. To test this hypothesis, we subjected protrusions generated in a period of 24 hrs by individual cells (**Fig. 4A**) that were found in pairs or in isolation (**Fig. 2Bii-iv**) to the Rayleigh test of directionality (see Methods). Cells were distinguished as pairs or as isolated based on the separation threshold for connectivity (**Fig. 8C**): cells with initial membrane to membrane separation $< 40\ \mu\text{m}$ were pairs, $> 40\ \mu\text{m}$ are isolated. **Figure 10** shows a representative collection of the angular distribution around a unit circle for individual cells. Both a cell in a pair and in isolation exhibit uniformity around the cell as seen by the low “ r ” values when subjected to the Rayleigh test (see Methods). **Table 1** shows that

Table 1. Rayleigh test results for individual cells experiencing different microenvironments (section 3.2.2).

	<i>Pairs</i>	<i>Isolated</i>	<i>Impermeable Boundary</i>
Number of Protrusions	920	1064	823
Number of Cells	20	28	15
Individual Cells Rejecting <i>Rayleigh's</i> H_0 with 95% Certainty	1	2	0

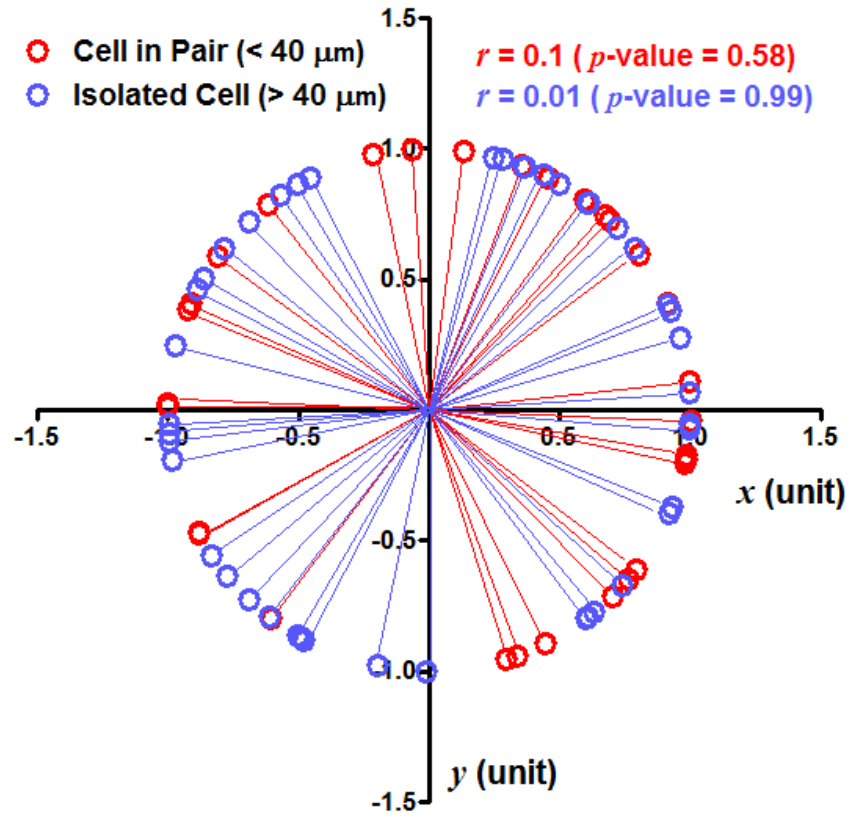


Figure 10. Angular distribution of protrusions originating from individual cells. xy -coordinates on the unit circle are shown for a cell that is found in a pair ($< 40 \mu\text{m}$ separation; red), and an isolated cell ($> 40 \mu\text{m}$ separation; blue). The resultant length, r , is 0.1 (p -value = 0.58) for cell in pair and 0.01 (p -value = 0.99) for isolated cell.

only 5% percent of the individual cells in pairs, 7% of the cells near found in isolation, and 0% of cells near an impermeable boundary ($< 40 \mu\text{m}$) rejected the null hypothesis for uniformity. The small fraction of cells that rejected Rayleigh's null hypothesis was subjected to the V-test (see Methods) without any of them rejecting the null hypothesis of net orientation of protrusions toward the direction of their neighboring cell (not shown). These results were not altered if the data was sorted into early ($< 12 \text{ hrs}$) and late ($\geq 12 \text{ hrs}$) times (not shown). In addition, the average protrusion length ($\sim 40 \mu\text{m}$) (**Fig. 11A**) and average protrusion frequency ($\sim 1.8 \text{ protrusions/hr}$) (**Fig. 11B**) were the same cells in pairs or in isolation. These observations altogether suggest that the protrusive activity of cells during early vasculogenesis is an unbiased, random process.

2-4 DISCUSSION

The set of experiments presented in this work were tailored to the study of vascular morphogenesis in a permissive microenvironment. This means for both angiogenesis and vasculogenesis studies presented here that: (1) no externally gradients of growth factors were imposed on any cell culture for guidance, (2) no dose-response of any signal (mechanical or chemical, loss or gain of function) were used to affect cells in culture, (3) individual ECs and the boundaries of the culture served as the only sources of perturbation to interaction, and (4) cells were allowed to arrange and interact with their microenvironment according to cell-dictated rules for network evolution. In the assays used here HUVECs interacted such that they spontaneously underwent vascular morphogenesis, either in the form of sprouting (Section 3.1) or vasculogenesis (Section 3.2) [9].

In the context of sprouting angiogenesis within this environment, we observed that cells responded to the presence of solid boundaries to which the collagen matrix was bound (**Fig. 6**).

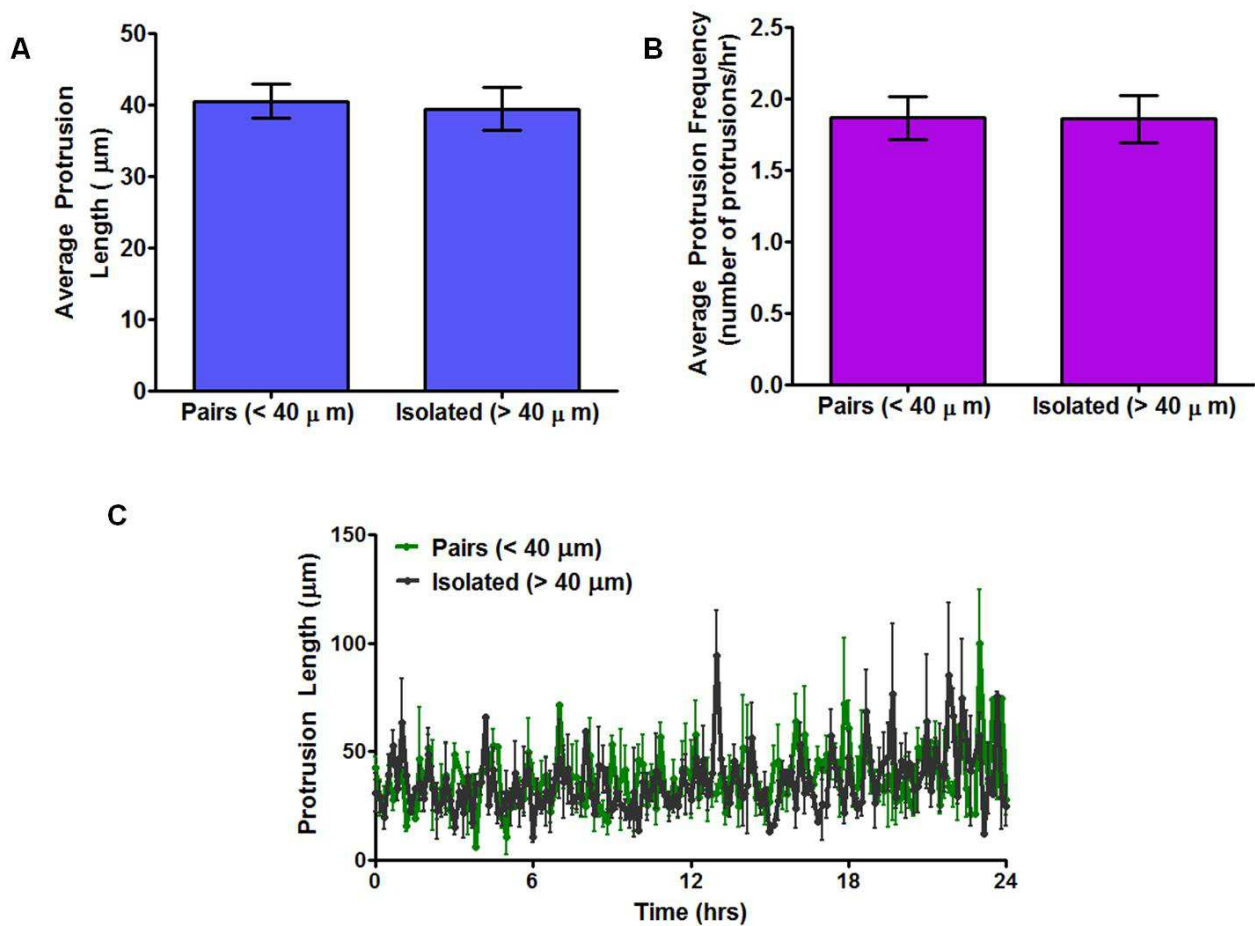


Figure 11. Characterization of HUVEC protrusions in 3D collagen scaffolds. (A) For every cell, the protrusion length was calculated from the cell center, and then cell radius was subtracted to determine protrusion length from membrane. An average of protrusion length for all cells in their respective categories is shown. No significant differences were found. (B) For every cell, the number of protrusions was determined and divided by the experimental time to determine the protrusion frequency of a cell. An average of protrusion frequency for all cells in their respective categories is shown. No significant differences were found. (C) Average protrusion length was tracked throughout an experimental time of 24 hrs. Cells less than 40 μm of initial separation = 20 (green); cells greater than 40 μm of initial separation = 28 (black).

With this observation, we were not able to discriminate if the response to boundaries was dominated by contact or non-contact (at a distance) communication. However, with the high density seen for 50 μm -diameter (corresponding to a single invasion in every well) we interpret that this well size is similar to the size of a luminal invasion. When we further segmented our culture wells into equal area annuli we found that the outermost annuli contained a significantly higher density of luminal invasions for both 4 mm- and 400 μm –diameter wells (**Fig. 7A, B**). With this spatial analysis, we identified that the direct contact to the wall explained the higher density of luminal invasions near boundaries. By calculating the fraction of lumens that are attached to the wall, we confirmed that the attraction of luminal invasions towards the wall was contact dependent; the majority of the luminal invasions near the boundary were directly attached to the boundary (**Fig. 7C**). Altogether, these observations suggest that action at a distance (soluble or mechanical signals, **Fig. 1**) is not playing a dominant role in the emergence of luminal invasions. The observations reported here are consistent with contact-dependent interactions of the ECs and their microenvironment: the preferential emergence of invasions exhibited physical attachments to the boundary. Additional considerations further strengthen this suggestion; we discuss them here.

Biomechanical stimuli often require contact: cells respond to mechanical substrates by establishing adhesion sites to the extracellular environment. The attachment of the collagen fibers onto the wall (see Methods, **Fig. 3D-E**) may serve as an anchor point which could translate to a region of high matrix stiffness for the cells. This consideration and the preferential emergence of invasions at the boundary (**Fig. 7**) would be in agreement with the reported observation that cells will preferentially migrate towards zones of higher matrix stiffness(durotaxis) [39]. Our results may be coherent with a mechanotactic response of

invasions since they were directly attached to a solid substrate while the opposing environment was very soft (0.3% collagen). The initial absence of collagen near the boundary could have facilitated the emergence of invasions at this interface. Since invasions were found physically adhered to the boundary (**Fig. 5**), we assessed the possibility that initially collagen was not present at the boundary (**Fig. 3, D-E**). Both, the bulk collagen and the collagen at the boundaries exhibited similar fiber structure prior to seeding the monolayer of cells; we do not attribute the boundary effect to heterogeneities in the architecture of the collagen but to the MT-MMP activity over time from the cells.

We used an approach analogous to our invasion angiogenesis studies to understand communication at the individual cell level relevant to the context of vasculogenesis (**Fig. 1B**). We first observed that individual cells are likely to establish connection if they are almost touching each other initially (**Fig. 8C**). A prominent behavior of cells on substrates is their migratory activity [39-40]. It has been suggested that the fundamental mechanism for cell-cell communication and network formation is the chemotactic migration of ECs that originates from paracrine VEGF signaling [11]. In addition, chemotactic response of ECs to externally applied gradients of VEGF has been documented [20]. We made a qualitative comparison between the migratory activity of cells on 2D substrates versus cells fully embedded in 3D collagen matrices; we observed a dramatic reduction in motility within 3D environments (**Fig. C4 Appendix C**), compatible with the observation that 3D fibrous environment presents a physical hindrance for cells to move freely [41]. Our quantitative analysis of EC migration showed that the movement of cells was highly constrained (**Fig. 9C**): over 24 hrs, they moved only ~10 μm while the characteristic separation for connection was 40 μm , suggesting that cell prepositioning may have an important role in pattern formation. This observation also suggests that migration is not

contributing to establishing connections (**Fig. 9**). In addition, cells in isolation (**Fig. 9C**) or in pairs (**Fig. 9D**) did not change their migration activity, suggesting that a potential paracrine signal [11] or mechanical perturbation [5] at a distance did not lead to a migratory response of ECs during early vasculogenesis in 3D collagenous environments. Surprisingly, protrusive activity was found to remain uniform around individual cells regardless of their proximity to neighboring cells or a boundary (**Table 1**). The absence of protrusion directional bias or polarity from ECs fully embedded in 3D collagen (**Fig. 10**), does not agree with the observations for individual cells in 2D [20] and for angiogenic tip cells [33, 42]. The lack of response in migratory activity and protrusive directionality are not compatible with our initial hypothesis: distant signals do not dominate the communication between individual ECs during early times of vasculogenesis.

Cellular traction that acts at a distance to guide cells to interaction has also been implicated as a fundamental mechanism that ECs use for communication [5]. We do not discard the possibility that mechanical signals played a role in the observed contact-dependent responses of ECs. Due to the intricate 3D network of fibers, the decay of the tractional field from a cell could be very short. In the potential use of mechanical communication, ECs would have to overcome the limitation of fast decay in tractional field by secreting additional ECM proteins or bundling the surrounding collagen [30]. The local increase in fiber concentration transduces traction faster at the expense of distance [27] (we note that we may not have been able to resolve the length scale on which such interactions existed). The lack of directionality on protrusive activity of cell pairs (**Table 1, Fig. 10**) suggests that the potential sensing role of such protrusions is not necessarily active in 3D. The secretion and or bundling of the ECM could also explain the nature of the protrusive activity of ECs at early times of vasculogenesis in 3D

environments. Further work is required to address the interplay between protrusive activity and morphology with cell-generated matrix remodeling. For instance, ECs could be transfected with GFP (**Fig. C-5** Appendix C) and ECM architecture can be observed concurrently using reflectance microscopy (**Fig. 3D-E, Fig. C-5** Appendix C).

Our results are compatible with a contact-dependent mechanism for EC communication that leads to connectivity, and consequently to network formation (**Fig. 8**). The separation threshold for connected cell pairs (**Fig. 8C**) is approximately the same as the average protrusion length of individual cells (**Fig. 11**); this suggests that protrusive activity may be random search for cell-cell contact. We note that ECs were able to establish connections in the absence of any biased activity (**Fig. C-6** Appendix C). A mathematical model was proposed where cells are able to generate patterns solely based on contact response [43]. To the best of our knowledge no experimental data has been reported that argues in favor of contact-dependent signal for pattern formation.

2-5 CONCLUSION

Working in 3D culture *in vitro*, our observations suggest that both invasion angiogenesis and vasculogenesis may be regulated by contact signals. Further work is required to elucidate the mechanisms that guide cell-cell communication in 3D environments. Particularly for vascular morphogenesis, manipulating the evolution of pattern formation may lead to successful tissue engineering constructs with perfusable microvasculature in the bulk. For instance, both vasculogenesis and angiogenesis could be combined in a single platform and allowed to progress concurrently. Using dense collagen to generate a microfluidic system [30, 44], the surfaces of channels could be lined with ECs while ECs are seeded in the bulk of the collagen. This

combination could lead to the connection of luminal sprouts with vessel structures that are within a collagen network; conferring continuity between outside flow and bulk of the culture. Such a system would open new opportunities to consider growing thick tissues *in vitro*.

2-6 REFERENCES

1. Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., Walter, P., ed. *Molecular Biology of The Cell*. 5th ed. 2008, Garland Science.
2. Matsuda, M., Koga, M., Nishida, E., Ebisuya, M., *Synthetic Signal Propagation Through Direct Cell-Cell Interaction*. Science Signaling, 2012. **5**(220): p. 1-8.
3. Liebner, S., Cavallaro, U., Dejana, E., *The Multiple Languages of Endothelial Cell-to-Cell Communication*. Arterioscler Thromb Vasc Biol, 2006. **26**: p. 1431-1438.
4. Betsholtz, C., et al., *VEGF guides angiogenic sprouting utilizing endothelial tip cell filopodia*. Journal of Cell Biology, 2003. **161**(6): p. 1163-1177.
5. Reinhart-King, C.A., M. Dembo, and D.A. Hammer, *Cell-Cell Mechanical Communication through Compliant Substrates*. Biophysical Journal, 2008. **95**(12): p. 6044-6051.
6. Risau, W. and I. Flamme, *Vasculogenesis*. Annual Review of Cell and Developmental Biology, 1995. **11**: p. 73-91.
7. Anderson, D.J., H.U. Wang, and Z.F. Chen, *Molecular distinction and angiogenic interaction between embryonic arteries and veins revealed by ephrin-B2 and its receptor Eph-B4*. Cell, 1998. **93**(5): p. 741-753.
8. Carmeliet, P., *Mechanisms of angiogenesis and arteriogenesis*. Nature Medicine, 2000. **6**(4): p. 389-395.
9. Folkman, J. and C. Haudenschild, *Angiogenesis in vitro*. Nature, 1980. **288**(5791): p. 551-6.
10. Davis, G.E., K.J. Bayless, and A. Mavila, *Molecular basis of endothelial cell morphogenesis in three-dimensional extracellular matrices*. Anatomical Record, 2002. **268**(3): p. 252-275.
11. Serini, G., et al., *Modeling the early stages of vascular network assembly*. Embo Journal, 2003. **22**(8): p. 1771-1779.
12. Manoussaki, D., et al., *A mechanical model for the formation of vascular networks in vitro*. Acta Biotheoretica, 1996. **44**(3-4): p. 271-282.

13. Weinstein, B.M., et al., *Endothelial tubes assemble from intracellular vacuoles in vivo*. Nature, 2006. **442**(7101): p. 453-456.
14. Helmlinger, G., et al., *Growth factors - Formation of endothelial cell networks*. Nature, 2000. **405**(6783): p. 139-141.
15. Szabo, A. and A. Czirok, *The Role of Cell-Cell Adhesion in the Formation of Multicellular Sprouts*. Mathematical Modelling of Natural Phenomena, 2010. **5**(1): p. 106-122.
16. Gerhardt, H., et al., *VEGF guides angiogenic sprouting utilizing endothelial tip cell filopodia*. Journal of Cell Biology, 2003. **161**(6): p. 1163-1177.
17. Hellstrom, M., Phng, L.K., Hofmann, J.J., Wallgard, E., Coultas, L., Lindblom, P., Alva, J., Nilsson, A.K., Karlsson, L., Gaiano, N., Yoon, K., Rossant, J., Iruela-Arispe, M.L., Kalen, M., Gerhardt, H., Betsholtz, C., *Dll4 signalling through Notch1 regulates formation of tip cells during angiogenesis*. Nature, 2007. **445**: p. 776-780.
18. He, Q.F., et al., *Improved cellularization and angiogenesis using collagen scaffolds chemically conjugated with vascular endothelial growth factor*. Acta Biomaterialia, 2011. **7**(3): p. 1084-1093.
19. Yin, Z.Z., et al., *Analysis of pairwise cell interactions using an integrated dielectrophoretic-microfluidic system*. Molecular Systems Biology, 2008. **4**.
20. Shamloo, A., et al., *Endothelial cell polarization and chemotaxis in a microfluidic device*. Lab on a Chip, 2008. **8**(8): p. 1292-1299.
21. Shamloo, A. and S.C. Heilshorn, *Matrix density mediates polarization and lumen formation of endothelial sprouts in VEGF gradients*. Lab on a Chip, 2010. **10**(22): p. 3061-3068.
22. Pedersen, J.A. and M.A. Swartz, *Mechanobiology in the third dimension*. Annals of Biomedical Engineering, 2005. **33**(11): p. 1469-1490.
23. Levenberg, S., et al., *Engineering vascularized skeletal muscle tissue*. Nature Biotechnology, 2005. **23**(7): p. 879-884.
24. Lampugnani, M.G., et al., *CCM1 regulates vascular-lumen organization by inducing endothelial polarity*. Journal of Cell Science, 2010. **123**(7): p. 1073-1080.

25. Davis, G.E., et al., *Coregulation of vascular tube stabilization by endothelial cell TIMP-2 and pericyte TIMP-3*. Journal of Cell Biology, 2006. **175**(1): p. 179-191.
26. Davis, G.E., W. Koh, and A.N. Stratman, *Mechanisms controlling human endothelial lumen formation and tube assembly in three-dimensional extracellular matrices*. Birth Defects Res C Embryo Today, 2007. **81**(4): p. 270-85.
27. Califano, J.P. and C.A. Reinhart-King, *A Balance of Substrate Mechanics and Matrix Chemistry Regulates Endothelial Cell Network Assembly*. Cellular and Molecular Bioengineering, 2008. **1**(2-3): p. 122-132.
28. Deen, W.M., *Analysis of Transport Phenomena*. 1998.
29. Davis, G.E. and C.W. Camarillo, *An alpha 2 beta 1 integrin-dependent pinocytic mechanism involving intracellular vacuole formation and coalescence regulates capillary lumen and tube formation in three-dimensional collagen matrix*. Experimental Cell Research, 1996. **224**(1): p. 39-51.
30. Cross, V.L., et al., *Dense type I collagen matrices that support cellular remodeling and microfabrication for studies of tumor angiogenesis and vasculogenesis in vitro*. Biomaterials, 2010. **31**(33): p. 8596-8607.
31. Mardia, K.V., *Directional Statistics*. 2006.
32. Batschelet, E., ed. *Circular Statistics in Biology*. Mathematics in Biology, ed. R.a.C. Sibson, J.E. 1981, Academy Press.
33. Chen, R.R., et al., *Integrated approach to designing growth factor delivery systems*. FASEB J, 2007. **21**(14): p. 3896-903.
34. Yannas, I.V., et al., *Synthesis and characterization of a model extracellular matrix that induces partial regeneration of adult mammalian skin*. Proc Natl Acad Sci U S A, 1989. **86**(3): p. 933-7.
35. Zheng, Y., et al., *Microstructured templates for directed growth and vascularization of soft tissue in vivo*. Biomaterials, 2011. **32**(23): p. 5391-5401.
36. Stratman, A.N., et al., *Endothelial cell lumen and vascular guidance tunnel formation requires MT1-MMP-dependent proteolysis in 3-dimensional collagen matrices*. Blood, 2009. **114**(2): p. 237-47.

37. Little, C.D., P.A. Rupp, and A. Czirok, *alpha v beta 3 integrin-dependent endothelial cell dynamics in vivo*. Development, 2004. **131**(12): p. 2887-2897.
38. Drake, C.J., et al., *VEGF regulates cell behavior during vasculogenesis*. Developmental Biology, 2000. **224**(2): p. 178-188.
39. Cheung, Y.K., et al., *Microscale control of stiffness in a cell-adhesive substrate using microfluidics-based lithography*. Angew Chem Int Ed Engl, 2009. **48**(39): p. 7188-92.
40. Poujade, M., et al., *Collective migration of an epithelial monolayer in response to a model wound*. Proc Natl Acad Sci U S A, 2007. **104**(41): p. 15988-93.
41. Stratman, A.N., et al., *Endothelial cell lumen and vascular guidance tunnel formation requires MT1-MMP-dependent proteolysis in 3-dimensional collagen matrices*. Blood, 2009. **114**(2): p. 237-247.
42. Shamloo, A., H. Xu, and S. Heilshorn, *Mechanisms of Vascular Endothelial Growth Factor-Induced Pathfinding by Endothelial Sprouts in Biomaterials*. Tissue Engineering Part A, 2012. **18**(3-4): p. 320-330.
43. Edelstein-Keshet, L., and Ermentrout, G.B., *Contact response of cells can mediate morphogenetic pattern formation*. Differentiation, 1990. **45**: p. 147-59.
44. Zheng, Y., et al., *In vitro microvessels for the study of angiogenesis and thrombosis*. Proc Natl Acad Sci U S A, 2012. **109**(24): p. 9342-7.

CHAPTER 3

ENGINEERING CELLULAR INTERACTIONS TO DRIVE SELF-ASSEMBLING OF MICROVASCULATURE IN VITRO

3-1 INTRODUCTION

Tissue engineering has the potential to solve clinical problems such as: skin section replacements, arterial or venous replacements in cardiovascular diseases, and even more ambitiously, the replacement of complete larger organs [1]. These solutions require large scale tissue constructs with: (1) appropriate physiology on both macroscopic and microscopic scales (which includes a suitable cell and biomaterial source), (2) proper maintenance during *in vitro* culture, and (3) compatibility with surgical implantation (e.g. surgical connections from engineered tissue to host). All of these considerations represent unmet challenges within the tissue engineering field.

Tissues require efficient mass transfer during *in vitro* culture and during post-surgical integration. However, the most challenging limitation of *in vitro* tissue cultures to date is the proper oxygenation and nourishment of the thick tissues of interest [2]. Standard culture systems provide purely diffusive delivery; this constraint has limited translation to the clinic to thin tissues such as skin [3-5] and vascular grafts [6-7] and tissues of low metabolic demand such as cartilage [8]. We and others have understood that part of the solution to this limitation is to induce the vascularization of scaffolds for tissue engineering [2, 9-11]. The oxygen depletion length (the distance over which oxygen can diffuse before being consumed; known as the Krogh length [11-12]) requires that capillary vessels in living tissues be within 100-200 μm of

one another in most tissues [13]. This requirement arises from the fact that distances in living tissues involve a high density of cells that actively consume oxygen.

Vascularizing tissues *in vitro* requires the construction of blood vessels, at this high density, over macroscopic distances. The efforts to accomplish this goal include: tailoring geometrical features on acellular biomaterials that accelerates cellular transport into the tissue [14], taking advantage of the self-organization of endothelial cells (ECs) or endothelial progenitor cells (EPCs) into vascular networks (vasculogenesis) in 3D matrices, and establishing co-cultures of ECs with specific tissue cell [15-18]. The works of Levenberg et. al. [15], and Koike, et. al.[18], deserve particular attention since both showed that when proper vascular tubes could be grown in 3D matrices and that the vasculature in these constructs connected (anastomosed) with host vasculature and became functional after implantation into mice. Their approach, based on non-perfused culture, limited the cell density and dimension of the cultures (~1mm thick, 3.2×10^7 cells/mL [10]), restricting the dimension of constructs to the order of their Krogh length [11-12]. The generalization of such approach to the engineering of thick tissues is limited: thick tissues are much larger than their Krogh length. In order to scale this effort to create highly cellularized and thick tissues, we need to establish a connection between an external source of blood or growth medium and a developing 3D capillary bed *in vitro*, before implantation.

Anastomosis represents the native mechanism to generate such connections *in vivo*. This process, by which an angiogenic sprout fuses with another sprout or pre-existing blood vessel, is still incompletely understood at both cellular and molecular scales. We know that tip cells of angiogenic sprouts migrate into open tissues (angiogenesis) and that signals such as vascular endothelial growth factor (VEGF) play a crucial role in their guidance [19]. The angiogenic

exploration of the tip cells guides stalk cells from the pre-existing blood vessel into the adjacent tissue; upon encountering other vessels, anastomosis can occur. However, we do not understand what an individual sprout does when it encounters another vessel. Song et. al., recently developed a PDMS microfluidic platform to assess the mechanisms that govern anastomosis [20]. They reported that *in vitro* angiogenic sprouts originating from opposing monolayers of ECs are able to anastomose leading to perfuseable vessel formation. This perfuseability of the vasculature *in vitro* is a highly desired feature for tissue engineering systems. However, the size and material of their system limits its applicability to growing tissues: (1) the health of a tissue could be compromised if a large tissue is studied using the same approach given that the vasculature would require a long time for anastomosis, (2) the ability of the vasculature to remodel the extracellular matrix is hindered by the host material (PDMS) from which their devices were formed, and (3) PDMS is not a biodegradable material suitable for integration with the host.

In **Figure 1A**, we present an approach to anastomose endothelial tubes at the capillary scale with larger vessels (**Fig. 1A**). In this platform, a monolayer of cells invades into the underlying collagen while cells inside a collagen scaffold form vascular conduits in a vasculogenesis model. We hypothesized that anastomosis would occur between the invasions from the monolayer and the tubes formed in the bulk (**Fig. 1**). To explore this hypothesis, we transduced a population of ECs with Red Fluorescent Protein (RFP) and a separate population with Green Fluorescent Protein (GFP) then studied the evolution of structure between ECs on a monolayer and ECs seeded in the bulk of collagen, and the distribution of the two cell populations. We demonstrate that these two processes lead to anastomosis by interdispersion and incorporation of the two cell populations into a self-assembled vascular architecture. We further

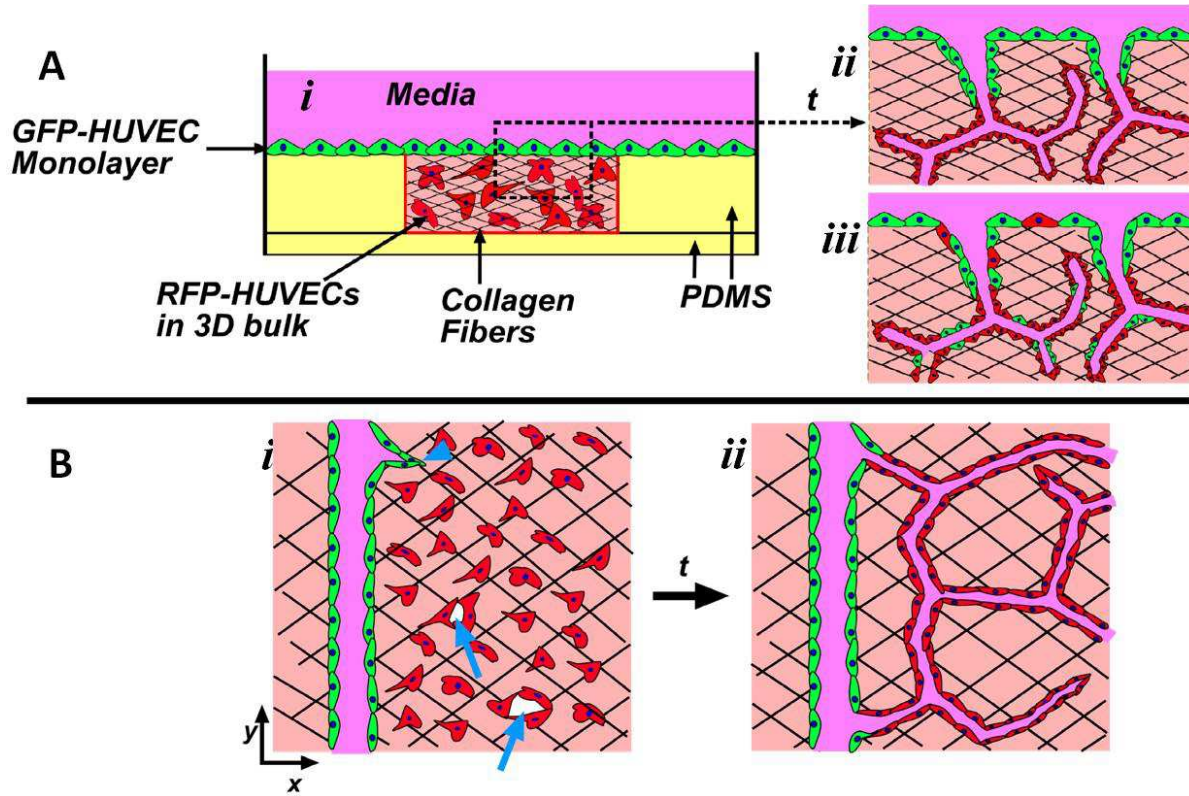


Figure 1. (A) Depiction of vertical anastomosis configuration. (Ai) 4 mm-diameter wells in PDMS contain seeded with red endothelial cells (ECs). The PDMS walls have been chemically treated as described to promote collagen adhesion (red delineated walls). Following gelation of cell-laden collagen, a monolayer of green ECs is seeded on the surface of the collagen. Growth medium (purple) with pro-angiogenic factors bathes the culture. (Aii, Aiii) After different time-points the monolayer cells invade the underlying collagen, the bulk cells remodel the surrounding collagen while undergoing vasculogenesis, and the two cell populations interdisperse within the endothelia on the surface and in the bulk. (ii) In the hypothesized interaction between cell populations, the monolayer cells and the bulk cells distribute sequentially. (iii) In the interaction between monolayer-bulk cells described in this work, monolayer cells are found intermixed with bulk cells. (B). Perfusable in Vitro Living Tissue (PiVLT). (Bi) The anastomosis interactions studied in the vertical format are extended to a microfluidic context. Channels lined with ECs represent monolayer cells that undergo invasion angiogenesis (blue arrowheads) into an underlying collagen that contains ECs undergoing vasculogenesis (blue arrows). (Bii) The anastomosis of vessels originating from vasculogenesis with vessels that are originating from angiogenic invasions establish a continuous path from one channel to the bulk of the collagen.

show that this process can be exploited within a microfluidic context, where monolayer cells form an endothelium that can be connected to flow inlets and outlets of a collagen microfluidic device (**Fig. 1b**). By designing microfluidic channels that are lined with ECs we mimic large-caliber vessels for convective transport over macroscopic distances. Connecting capillary-sized vessels to the large-caliber vessels is progress towards the creation of a Perfuseable In Vitro Living Tissue (PiVLT).

3-2 MATERIALS AND METHODS

3-2.1 Reagents

The following cells, reagents, and materials were used: Human umbilical vein endothelial cells (HUVECs), 1x Medium 199 (1x M199), fetal bovine serum (FBS), L-glutamine, penicillin/streptomycin, trypsin/EDTA, HEPES buffered saline solution (Lonza); endothelial cell growth supplement (ECGS), basic fibroblast growth factor (bFGF), phosphate buffered saline (PBS) (Millipore); vascular endothelial cell growth factor (VEGF) (R&D Systems); rat tails (Pel-Freez Biologicals, Rogers AR); phorbol-12-myristate-13-acetate (TPA), (Cell Signaling Technology, Danvers, MA); (poly)ethylenimine (PEI)(MW 750,000)(Sigma-Aldrich, St. Louis, Missouri); heparin solution, Alexa Fluor 488 phalloidin, Alexa Fluor 568 phalloidin, 4',6-diamidino-2-phenylindole (DAPI), 10x Medium 199 (10x M199) (Invitrogen); Acetic Acid (Mallinckrodt Chemicals); L-ascorbic acid (Acros Organics, Atlanta, GA); Bovine Serum Albumin, Triton X-100 (MP Biomedicals, Inc., Solon, OH); (poly)-dimethylsiloxane (PDMS) (Sylgard[®] 184, Dow Corning, Midland, Michigan); glass coverslips (Fisher Scientific); Sodium Hydroxide (NaOH) (VWR); Biopsy Punches (Miltex by Kai); Glutaraldehyde (Fluka); Formaldehyde (Polysciences, Inc.).

3-2.2 Cell Culture maintenance, fluorescent protein transduction and collagen stock solution

HUVECs were used in passages 3-9. For expansion and maintenance, cells were cultured in M199 containing 20% FBS, 30 µg/mL of ECGS, 5 Units/ml of heparin solution, 200 mM of L-glutamine, and 120 U/ml of penicillin/streptomycin. This growth media (GM) was changed every 2-3 days. **Human Brain Vascular Pericytes (PC)** were used in passages 2-5. PCs were cultured as described in vendor's instructions and using their media recommendations.

A cell batch was transduced with a virus containing a red fluorescent protein (RFP) sequence that is expressed constitutively inside the cell. The virus was added to the media for 24 hrs. GM was used to replenish the culture as it was kept under expansion. Another batch was transduced with GFP-virus in a similar fashion. The viral constructs were created in the School of Veterinary Medicine of Cornell University by Christian Abratte. To produce viruses, the construct was co-transfected with packaging (psPAX2, addgene#12260) and envelope (pmd2.g, addgene #12259) plasmids into HEK293T cells (100mm plate 50% confluent) using Mirrus Transit-LT1 transfection reagent, according to the manufacturer's instructions. Approximately 16 hrs after transfection, fresh media was added to the 293T cells and they were cultured for 72 hrs, during which, media was collected and refreshed, then pooled and concentrated using a 30kd MWCO centrifugal spin column. Concentrated media containing lentiviral particles was aliquoted and stored frozen at -80°C. The FUW-cherry construct was made by using the Cold Fusion recombineering kit to replace the eGFP in plasmid FUGW (addgene# 14883) with mCherry which was cloned from some other mammalian expression vector.

Collagen was extracted from rat tails as described previously [9]. Acidic collagen solution was then lyophilized, and the dry mass was determined. The collagen was resuspended in 0.1% v/v acetic acid to 8 mg/mL stock solution.

3-2.3 Preparation of suspension of cells in collagen

Cells at 75-95% confluency were washed with HEPES buffered saline solution, removed from culture flasks with trypsin-EDTA, neutralized with GM, centrifuged, and resuspended to a density of 3×10^6 cells/mL. Cell seeded collagen gels were prepared by mixing 10x M199 (1:10; compared to the desired final mixture volume), NaOH, 1N (1:50, compared to the required collagen stock volume), 1x M199 (as the last component added to complete final mixture volume), resuspended HUVECs, and stock collagen solution. Both collagen solution and culture molds were kept on ice while preparing the suspension. NaOH, 10x M199, 1x M199, and collagen were mixed prior to adding cells in order to neutralize the collagen. The final pH of the solution was about 7.4. The working concentration of collagen was 6 mg/mL and the density of GFP-HUVECs was $0.3 - 2 \times 10^6$ cells/mL. A volume of 1 mL of GM containing 50 μ g/mL L-ascorbic acid, 50 ng/ml TPA, 40 ng/ml VEGF, and 40 ng/mL bFGF was added to each well. We call this formulation Vasculogenesis Media (VM). Wells bathed with VM were placed in the incubator for 7 hrs.

3-2.4 Preparation of PDMS wells for vertical anastomosis

Figure 2 depicts the process steps involved in making PDMS wells for the vertical anastomosis assay. Uncured PDMS was poured onto a plexi-glass mold with an array of cylindrical posts 2-mm high and 4-mm in diameter. The mold had additional height to make a 2-mm thick base for each well. The mold with uncured PDMS was placed in an oven at 60°C for 2 hrs. Wells were separated from the master mold. Individual wells were separated from the array using a biopsy punch with an 8-mm diameter opening. An individual well of PDMS (**Fig. 2C**) was placed in the center of a well in a 6-well plate. PDMS wells positioned in 6-well plates were

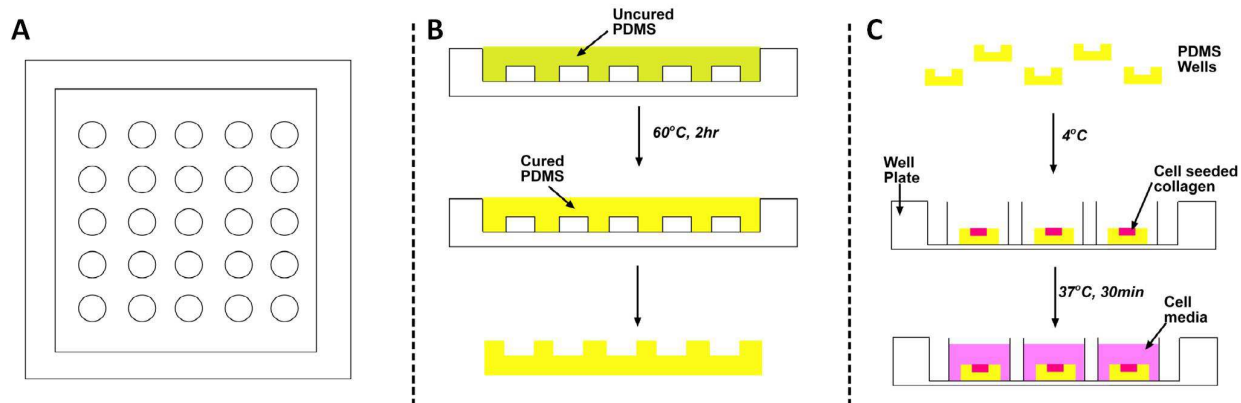


Figure 2. Generating PDMS wells for vertical anastomosis assays. (A) A plexi-glass mold formed by machining. (B) Uncured PDMS is molded onto the posts and cured. (C) Individual wells are punched out of the array of wells, placed in a standard well plate, seeded with cells and bathed in cell media.

kept at 4°C before being loaded with cell-seeded (GFP-HUVECs) or pure collagen. Cell-seeded or pure collagen were delivered to each well, a flat PDMS slab was positioned on top of the PDMS well, and the collagen was gelled inside an incubator for 30 mins at 37°C.

3-2.5 Culture in wells

Following gelation of collagen at 37°C for 30 mins, a volume of cell suspension in GM was allowed to sit on the surface of the collagen filled PDMS well for 3 hrs inside an incubator. This volume of cell suspended media is predetermined to deposit a cell concentration of 250cells/mm² of RFP-HUVECs on the surface of the entire well. VM was used to bathe the vertical anastomosis assay. VM was replenished every 2 days.

3-2.6 Fabrication and assembly of PiVLT

The surfaces of the top plexi-glass piece (**Fig. 3Aiii**) and the cover-slip (**Fig. 3Av**) were Activated as previously explained [9, 21]. Briefly, surfaces were treated with oxygen plasma, dip-coated with polyethylene amine (PEI), and allowed to react with glutaraldehyde (GA). These steps lead to covalent binding of collagen to the surfaces. The activated 25 mm × 25 mm cover-slip was placed in the slot of the bottom plexi-glass assembly part (**Fig. 3Avi, 4D**). A PDMS stamp (**Fig. 3Aiv**) with protruding features of two parallel channels was used to cast onto the collagen. The PDMS stamp and the top plexi-glass part were contacted with the ports of the plexi-glass aligned with the channel inlets of the stamp. Pins (**Fig. 3Ai**) were used to prevent obstruction of the ports by the injection of collagen. Using a 1mL syringe (**Fig. 3B**), the collagen/cells mixture was injected into the gel-loading port until it came out through the gel loading port (~300 - 400 µL in total, **Fig. 3D**). A volume of 170 µL was placed onto the center

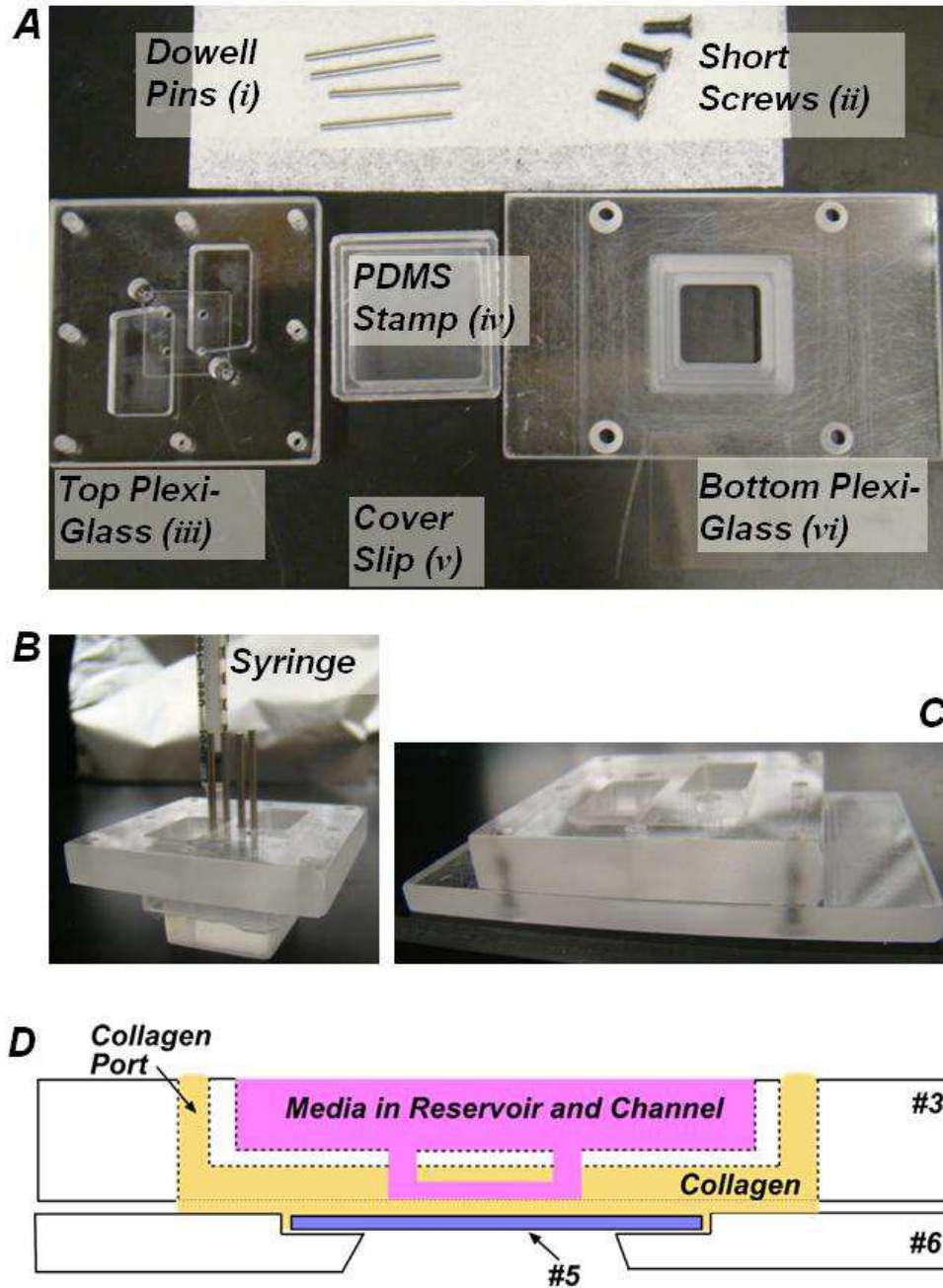


Figure 3. Perfuseable In Vitro Living Tissue (PiVLT) assembly depiction. (A) The different components used to setup the PiVLT. A 25mm×25mm cover slip is reclined onto the PDMS stamp (v). The Bottom PlexiGlass part (vi) has a thin trench to accommodate the cover slip. (B) To create microfluidic features in collagen, the collagen is injection-molded into a setup that constitutes the PDMS stamp (Aiv) pressed onto the Top PlexiGlass (Aiii) and the Dowell Pins (ii) inserted into the appropriate ports. The syringe has a specialized port. (C) The final assembly has the Bottom PlexiGlass (vi) with the collagen-coated coverslip (v) inserted in the trench, screwed onto the Top PlexiGlass (iii). (D) Cross-sectional schematic of one channel in an assembled device. One reservoir serves as inlet and outlet for a single channel (Aiii, Diii).

cover-slip on the bottom plexi-glass part. A slab of flat PDMS was positioned on top of this collagen solution in order to flatten the collagen. PDMS were exposed to O₂ plasma for 4 mins before contacting them with collagen solution to sterilize and increase wettability. Both the top and bottom layers of cell-laden collagen pieces were placed inside the incubator for 25 mins. Both PDMS parts were removed carefully, and the two pieces containing the polymerized collagen were placed together to create a 4-wall collagen channel (**Fig. 3C**). A volume of 10-20 μ L from a suspension of HUVECs at a density of 6×10^6 cells/mL was then injected into each channel to line the collagen walls of the fabricated device. Cells were left to attach for a minimum of 2 hrs followed by the removal of unattached cells with cell-free GM. Two medium reservoirs were machined into the top plexi-glass piece (**Fig. 3Aiii, 3C**). Each medium reservoir covered the inlet and outlet of a single (**Fig. 3D**) channel in a two-channel device in order to inhibit a pressure driven flow through the channel. Both reservoirs were filled to the same height with 1 mL of media.

3-2.7 Culture in PiVLT

Following monolayer cell attachment to collagen walls, fresh VM was added to the reservoirs (**Fig.3C,D**). The VM was added through one port inlet and allowed to flow through the channel length until the both ports are filled, which then they share the same volume of media through the same reservoir. A slow leakage from the vessels provided an interstitial flow that nourishes the macroscopic distances of the device. The leakage occurred between parts 3 and 6 of the assembled device (**Fig. 3D**). VM was replenished every 2 days (the leakage is slow enough that media is still present in the reservoir after this period).

3-2.8 Microscopy and Image Analysis

Stacks of images of horizontal sections (567 μm by 567 μm) were acquired at 3 μm -intervals in the vertical direction using confocal microscopy (Zeiss 710). A water immersion 25 \times Carl Zeiss objective with a numerical aperture of 0.8 was used. The normalized pinhole was 28 μm and the scan zoom was set to 0.6. The end position used to collect a stack was varied but was always greater than 150 μm . Every well was taken to represent an independent experiment. Three locations away from the wall were randomly selected for a given culture well, and the average of stacks acquired at these positions in each well was reported for as an independent experimental value (average of the normalized intensities, with respect to the total intensity in a stack, over the three positions at each depth – see **Eq. 1** for normalization). A custom Matlab® code (see Appendix D-5) was created to separate the red and green channels from the collected image stacks and perform subsequent analyses (**Fig. 4**). To extract intensity distributions through the depth of the culture, $f_{\alpha}(z)$ the total intensity of each color channel, α , was calculated in each slice and normalized with respect the total intensity in that channel in the entire stack:

$$f(z) = \frac{\sum_m I_z(m)}{\sum_z \sum_m I_z(m)}, \quad (1)$$

where m is the index of the pixels and n_z is the index of the slices. The slice $z = 0$ of the stack of images was defined as the slice with highest intensity in the red channel which represented the monolayer.

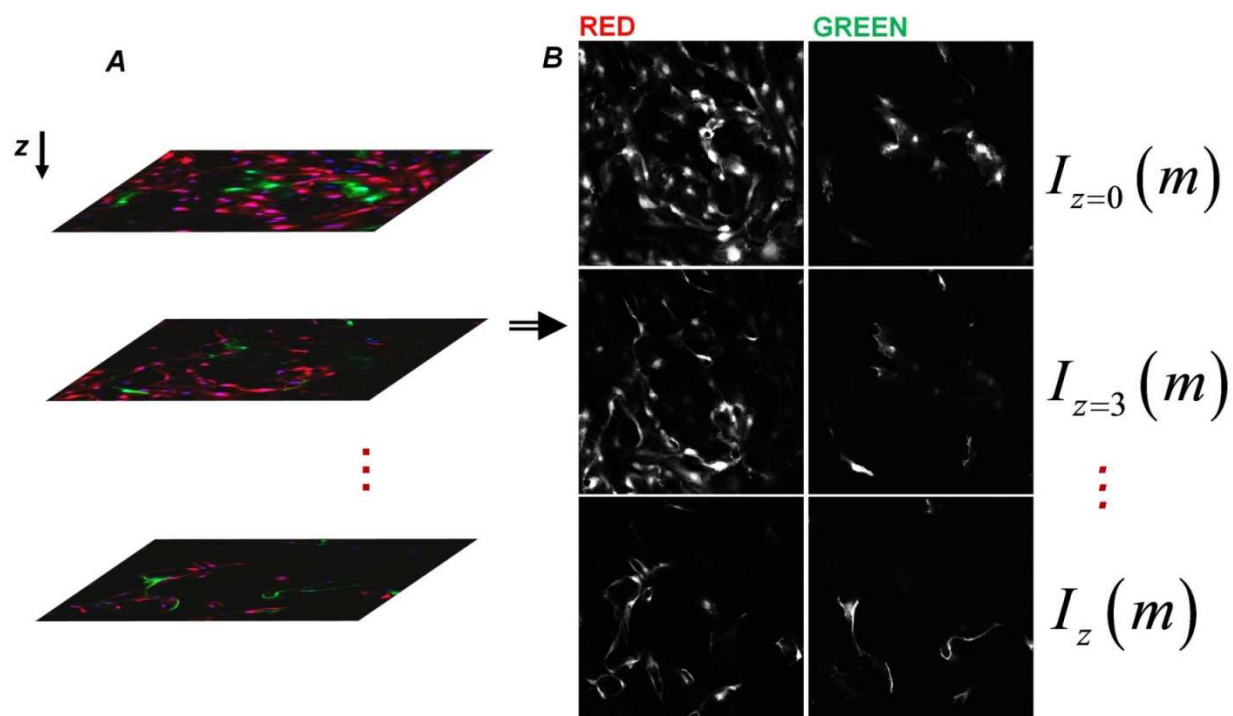


Figure 4. Analysis method for vertical anastomosis assay. (A) Stacks of 3μm-spaced images were collected using confocal microscopy. Every well represents an independent experiment. Three random non-overlapping locations within a well were used for averaging. (B) A custom Matlab® code was created to separate the red and green channels from the collected image stacks. The total intensity of an image (I) is tracked along z-direction to characterize the distribution of red and green structures. m is the linear index of the intensity matrix elements.

3-2.9 First moment and error propagation calculations

To characterize the depth distributions we calculated the depth-weighted average or first moment of the intensity distribution as follows:

$$\mu_1 = \sum z f(z). \quad (2)$$

We calculated error bars for the moments of the distributions following a propagation of error [22] through **Eq. 1**. We assumed an error in the “ z ” variable of 1 μm to account for the uncertainty in the physical location of the objective and stage of the microscope. We further assumed that the uncertainties in z and $f(z)$ were independent. We applied the rule of error propagation for multiplication using **Equation 3** and **Equation 4**.

$$\sigma_{z_i f(z_i)} = z_i f(z_i) \sqrt{\left(\frac{\sigma_{z_i}}{z_i}\right)^2 + \left(\frac{\sigma_{f(z_i)}}{f(z_i)}\right)^2} \quad (3)$$

$$\sigma_{f(z_i)} = \sqrt{\frac{1}{N} \sum_{r=1}^N (f_a(z_i) - f(z_i))^2} \quad (4)$$

Finally we applied the addition error propagation rule for Eq. 2 using **Equation 5**.

$$\sigma_{\mu_1} = \sqrt{\sum_i (\sigma_{z_i f(z_i)})^2} \quad (5)$$

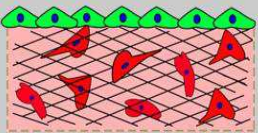
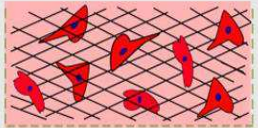
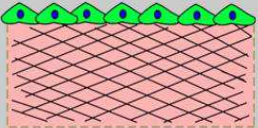
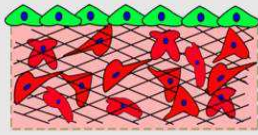
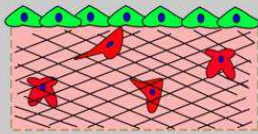
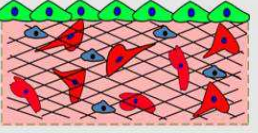
3-3 RESULTS AND DISCUSSION

3-3.1 Monolayer cells anastomose and mix with bulk cells

In our angiogenesis model, multi-cellular lumenized sprouts emerge from the planar configuration of monolayer cells (MCs) into the underlying collagen while in our vasculogenesis model cells inside the bulk (BCs) of a collagen scaffold undergo tubulogenesis and fusion to form vascular conduits (**Fig. 1Ai**) [9, 23]. We hypothesized that anastomosis could be achieved by the combination of the two steps and that the interactions between the two cell populations would be sequential (**Fig. 1Aii**). To test this hypothesis, we used a configuration where MCs were seeded on collagen at a surface density of 250 cells/mm²(RFP-HUVECs) to generate lumen-containing angiogenic invasions, and BCs were seeded at a density of 1×10⁶ cells/mL (GFP-HUVECs) to form a connected capillary network by vasculogenesis; this is our Standard Culture (**Table 1-1**) [24]. **Figure 5A** presents fluorescent micrographs that show the qualitative features observed in these experiments. We first note the following: at the monolayer, lumenized extensions invaded the collagen (**Fig. 5A**, $z = 0 \mu\text{m}$, stars); and in the bulk, vessel structures formed (indicated by the open lumen structures), as expected (**Fig. 5A**, $z > 0 \mu\text{m}$, arrows and arrowheads). These observations indicate that sprouting and vasculogenesis can occur simultaneously, perhaps unsurprisingly based on the robustness of both of these phenomena in 3D cultures of HUVECs [9, 23].

We further note in **Fig. 5A** that lumens emerging from the monolayer appear to have connected via anastomosis with lumens in the bulk by vasculogenesis. In support of this conclusion, we note the cross-sectional view in **Fig. 5B** shows the progression of luminal space from the monolayer into the bulk to depths over 500 μm . Comparison of control cases with BCs only (**Table 1-2 - Fig. 5C**) or MCs only (**Table 1-3; Fig. 5D**) suggests that the large vessel structures were not simply the result of extensive vasculogenesis or pure anastomosis. The presence of both BCs and MCs within these tubes further supports the idea that fusion occurred

Table 1. Summary and description of cellular conditions examined for anastomosis

CONDITION	CODE NAME	DESCRIPTION	DEPICTION OF INITIAL CONDITION
1. Standard culture	1E6	Monolayer (MC), at a surface density of 250 cells/mm ² and bulk cells (BC), at a density of 1x10 ⁶ cells/mL, interacted for 7 days from culture setup	
2. Bulk cells only	BC-Only	BC interacted with collagen for 7 days from culture setup	
3. Monolayer cells only	MC-Only	MC interacted with collagen only for 7 days from culture setup	
4. High Bulk Density	2E6	MC and BC, at a density of 2x10 ⁶ cells/mL, interacted for 7 days from culture setup	
5. Low Bulk Density	3E5	MC and BC, at a density of 3x10 ⁵ cells/mL, interacted for 7 days from culture setup	
6. Incorporation of Pericytes in bulk	PC	MC and BC interacted with pericytes incorporated in the bulk (1:5, PC:EC) for 7 days from culture setup	



GFP-HUVEC



RFP-HUVEC



Pericyte

between tubes formed from the monolayer and within the bulk. We are unaware of any previous report of anastomosis of tubes formed by these two distinct processes; this simple experiment provides a new basis for studying anastomosis *in vitro* and could open a new route to the formation of extensive networks of capillaries in engineered tissues.

In contrast to our hypothetical mode of interaction between MCs and BCs (**Fig. 1Aii**), the lumenized invasions presented heterogeneous composition with respect to the identity of cells: at the monolayer, invasions were predominately composed of MCs (**Fig. 5A**, $z = 15\ \mu\text{m}$, star and arrow, red); in the bulk, tubes were formed of purely of BCs (**Fig. 5A**, $z = 45\ \mu\text{m}$, arrow, green) and of combination of MCs and BCs (**Fig. 5A**, arrowheads). These images indicate that sprouts originating from the monolayer anastomosed with vessel structures inside the underlying collagen providing luminal access to the inside vasculature. Live imaging of this process would further enhance the characterization of the dynamics of connectivity and luminal extension.

3-3.2 Migration of monolayer cells into the bulk

The intermixing of the two cell populations was an unanticipated feature of the interactions during the anastomosis process. As mentioned above, sprouting angiogenesis is understood as a collective migration of cells originating from an endothelium and is concerted by a tip cell. The ability of individual MCs to rapidly incorporate into and migrate along endothelial structures in the bulk was a striking observation. As shown in Chapter 2 of this thesis, individual ECs in 3D collagen do not migrate more than their size, therefore it is unlikely that MCs would have migrated individually into the bulk before any structure was formed in the bulk. We sought to understand how the inclusion of ECs in the bulk influenced the process of cellular invasions during sprouting angiogenesis. To quantify the distribution of the cellular structures along the

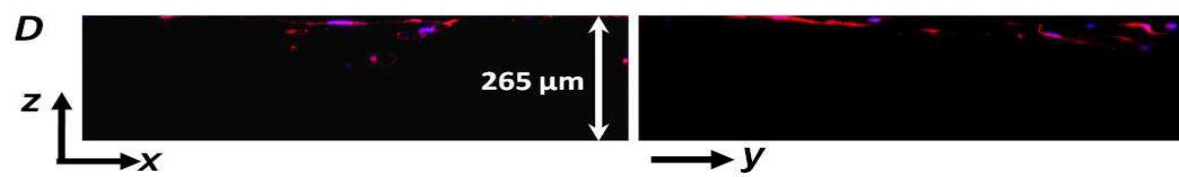
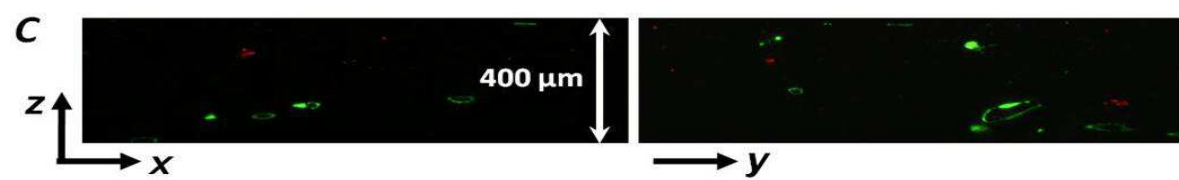
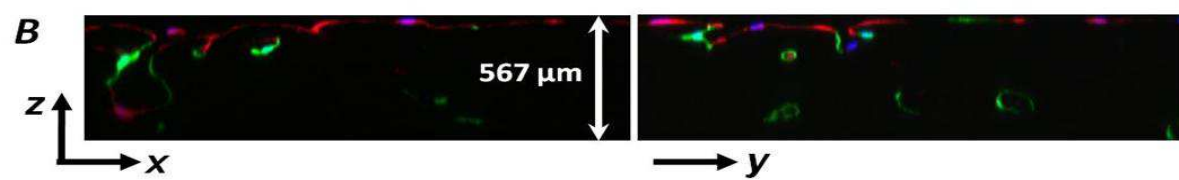
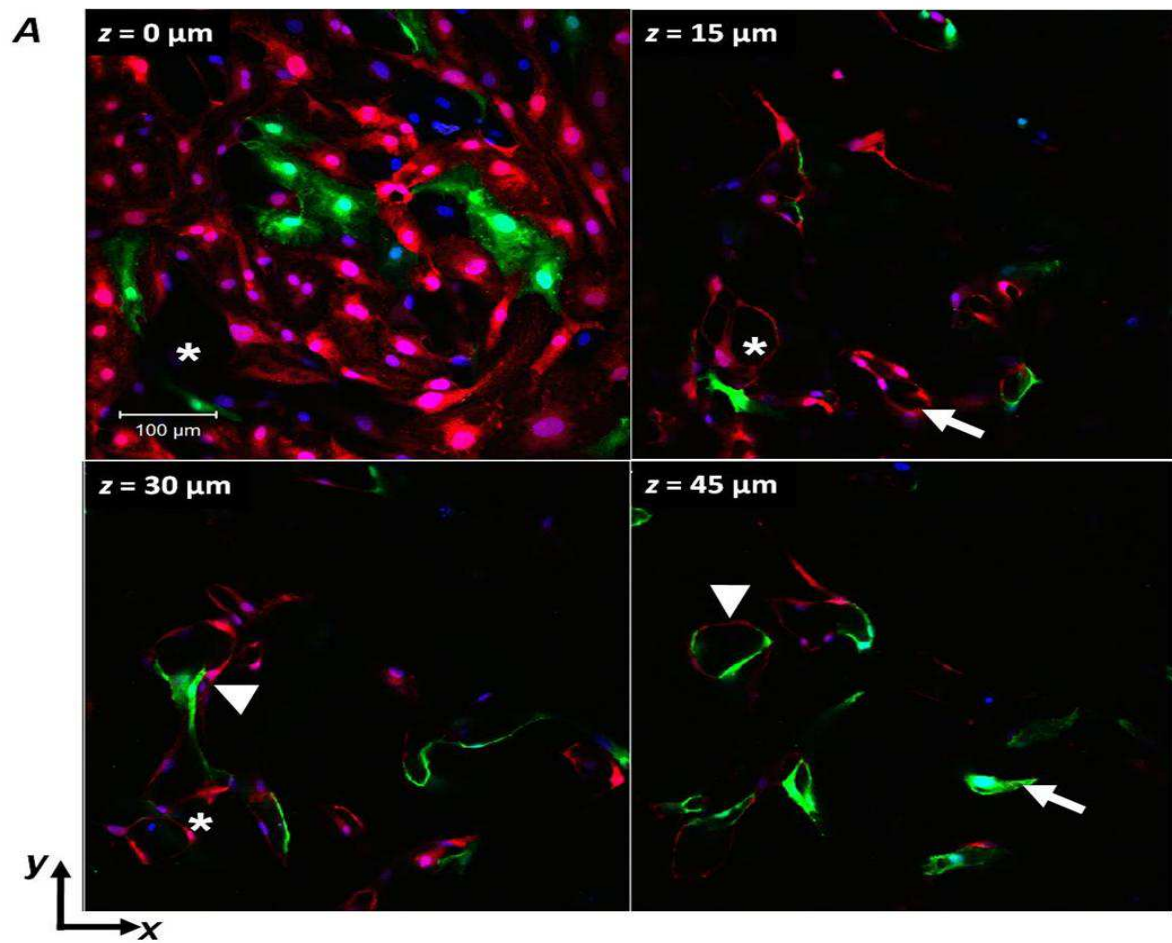


Figure 5. Fluorescence confocal images showing characteristics of vertical anastomosis experiments after 7 days. (A) A monolayer of RFP-HUVECs ($z = 0\mu\text{m}$) invades (white star; the empty structure found at $z = 0\mu\text{m}$ is found with multi-cellular structures at $z = 15\mu\text{m}$, and $z = 30\mu\text{m}$) and interacts with GFP-HUVECs from the bulk of the collagen gel. GFP-HUVECs become part of the monolayer. Mixed cell invasions (white arrows in frames $z = 15\mu\text{m}$, and $45\mu\text{m}$), as well as non-mixed cell invasions are found (white arrowheads in frames $z = 30\mu\text{m}$ and $45\mu\text{m}$) in the collagen. (B) ZX-view and ZY-view of stack segmented in (A). The depth of the z-stack is $567\mu\text{m}$. (C) ZX-view and ZY-view of stack from control case of bulk cells only. The length of the z-stack is approximately $400\mu\text{m}$. (D) ZX-view and ZY-view of stack from control case of monolayer cells only. The length of the z-stack is approximately $265\mu\text{m}$.

depth of the culture, we collected stacks of images starting from the endothelium and progressing into the depth of the matrix (**Fig. 4**). To characterize the distribution of red and green cells in a stack, we calculated the total intensity of red (for RFP-HUVECs) and green (for GFP-HUVECs) in each image within the stack; we normalized the intensities of an image with respect to the total intensity of the stack (see methods). Standard cultures (**Table 1-1**) with endpoints of 3, 5, and 7 days were studied to characterize the progression of cells as they migrated. At 3 days of culture (**Fig. 6A**, black), the presence of MCs at the monolayer interface ($z = 0 \mu\text{m}$) was found to be highest compared to 5 and 7 days (**Fig. 6A**, blue and purple, respectively). This trend suggests that MCs progressively left the monolayer (lowering of red signal at $z = 0 \mu\text{m}$) and penetrated further into the collagen at later days (**Fig. 6A** inset, compare black with purple, 3 and 7 days, respectively). Conversely, for the cells originally seeded into the bulk, the intensity at the surface rose with time (**Fig. 6B**). We associate this rise with the incorporation of BCs into the monolayer, as is seen in **Fig. 5A**. Beyond a depth of $25 \mu\text{m}$, the distributions of BCs remained similar at different culture times (**Fig. 6B**, and inset): oscillations in intensity along the depth of the culture represent the random presence of structures arising due to vasculogenesis (**Fig. 5B**).

To further characterize the evolution of these distributions of the two populations of cells, we calculated the first moments (or intensity weighted average depth; see methods) of the distributions of intensity (**Fig. 6C**). We performed a one-way ANOVA test, followed by Tukey's Multiple Comparisons Test using a 95% CI on moments for MCs (red) and independently for BCs (gray). The tests identified significant changes in the average depth between time-points for MCs; no significant changes occurred in the average depth of BCs even when compared to the BC-only control case (**Fig. 6C**, Day 7). More MCs were found deeper into the bulk as a function

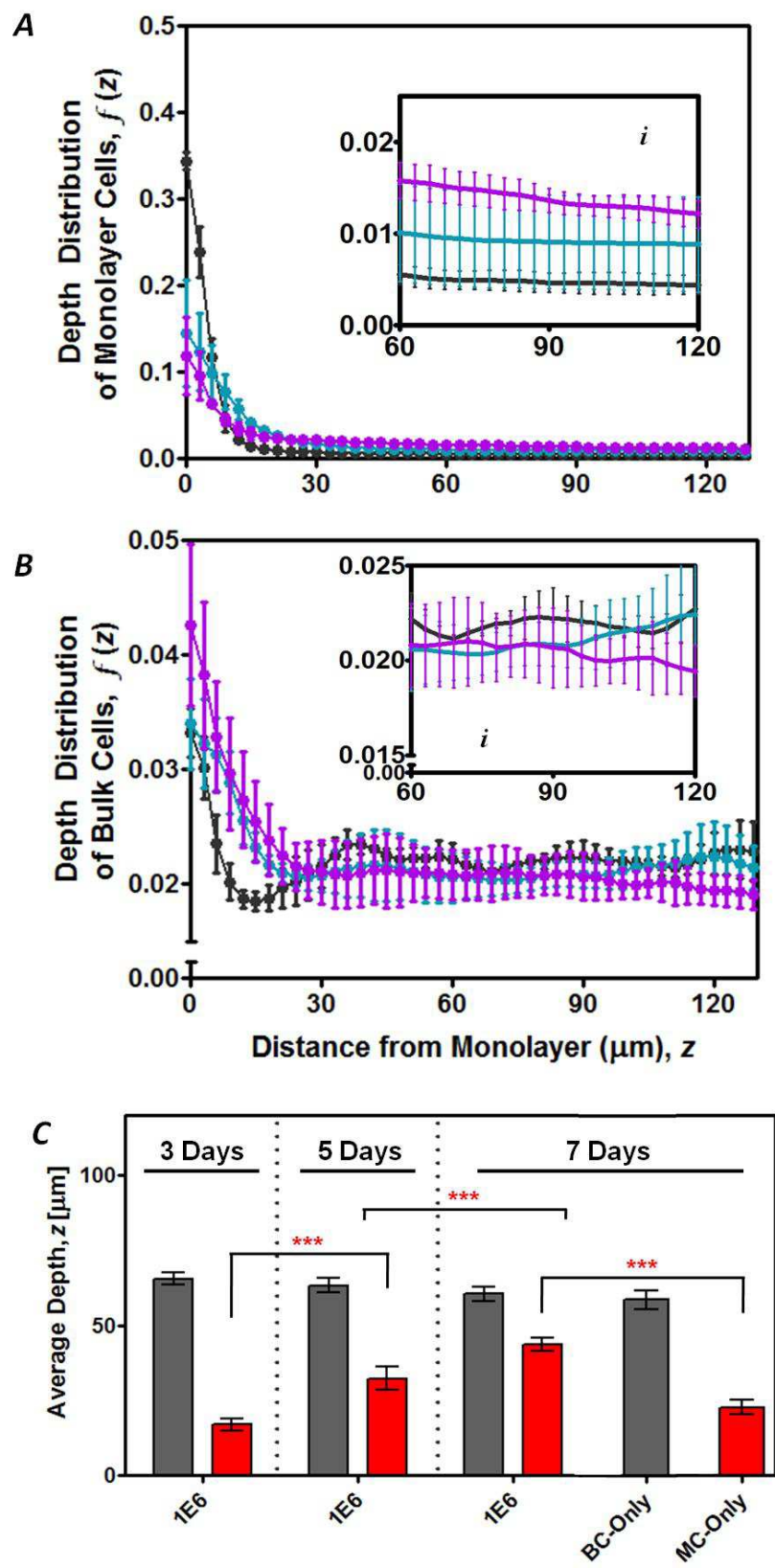


Figure 6. Characterization of the depth distribution of cell populations with time. (A-B) Distribution of the fluorescence intensity in the depth measured from the position of the monolayer. Cells initially seeded on the surface of the gel (RFP-HUVECs) are shown in (A); cells initially distributed within the bulk (GFP-HUVECs) of the gels are shown in (B). Cells in the bulk were seeded at a density of 1×10^6 cells/mL; monolayer cells at 250 cells/mm². Time-points of 3 (black), 5 (blue), and 7 (purple) days are shown. (C) Intensity weighted average depth. The average depth for monolayer cells in a standard anastomosis assay are in red; for bulk cells, gray.

of the time spent in culture. These results confirm the qualitative observation of inter-migration of MCs into the bulk (**Fig. 5**).

3-3.3 Different cellular conditions: network robustness and pericyte stabilization

We found that MCs intermixed with BCs in vessel structures deep within the culture well (**Figs. 5 and 6**). We became interested in understanding whether additional cellular conditions (**Table 1**) could impact the ability of MCs to anastomose with BCs and their migration. The additional conditions considered consisted of: (1) a high (2×10^6 cells/mL) bulk EC density (**Table 1-4**), (2) a low (3×10^5 cells/mL) bulk EC density (**Table 1-5**), and (3) the incorporation of pericytes with the bulk ECs (**Table 1-6**).

We first explored the idea that the depth to which MCs migrate would depend on the concentration of BCs in the bulk. We tested this hypothesis by exposing the MCs to a collagen containing a higher (2×10^6 cells/mL; **Table 1-1**) or lower (3×10^5 cells/mL; **Table 1-5**) BC density, compared to the standard case (2×10^6 cells/mL; **Table 1-4**). Previous experiments in our lab suggest that a higher BC density will show a robust capillary network while the lower BC is below the percolating threshold [24] for network formation. These conditions were compared against a MC-only case (**Table 1-3; Fig. 7A**). Qualitatively, we found that a reduced presence of MCs intermixed with BCs at low BC density (**Fig. 7B**) while strong association was observed between MCs and BCs at higher BC density (**Fig. 7C**).

In **Figure 8**, we present the first moments of the distributions of intensity associated with the MCs (**Fig. 8A**) and the BCs (**Fig. 8B**) for these conditions at the day 7 time point. As expected, quantification of the distribution in these cases revealed that MCs migrated significantly deeper in the 2×10^6 case compared to the 3×10^5 case (**Fig. 8A**, purple and white,

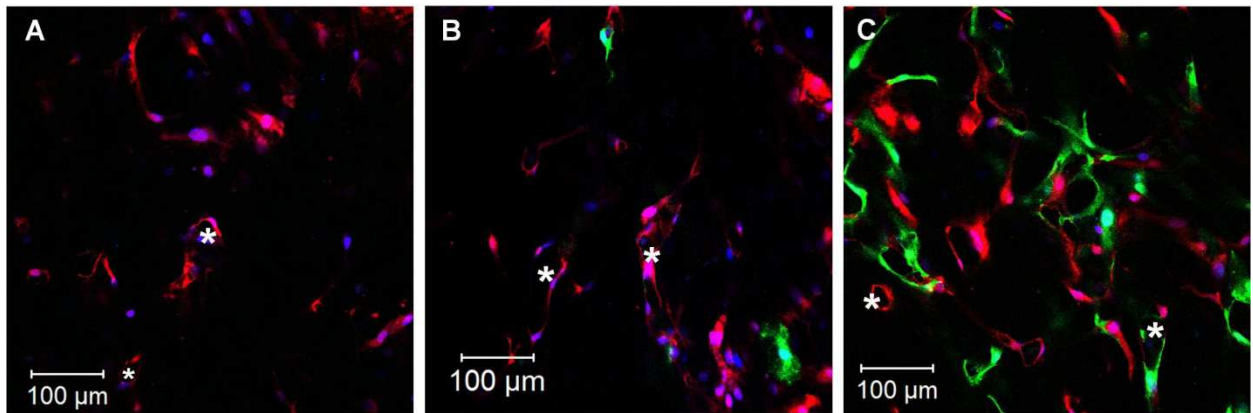


Figure 7. Culture quality with variations in the bulk cell density. RFP-HUVECs constitute the monolayer cells and GFP-HUVECs the bulk cells. A confocal image at $z = 15\mu\text{m}$ is shown for (A) MC-only, (B) 3×10^5 and (C) 2×10^6 . Clear examples of sprouts from endothelium invade with luminal structures (A-C, stars).

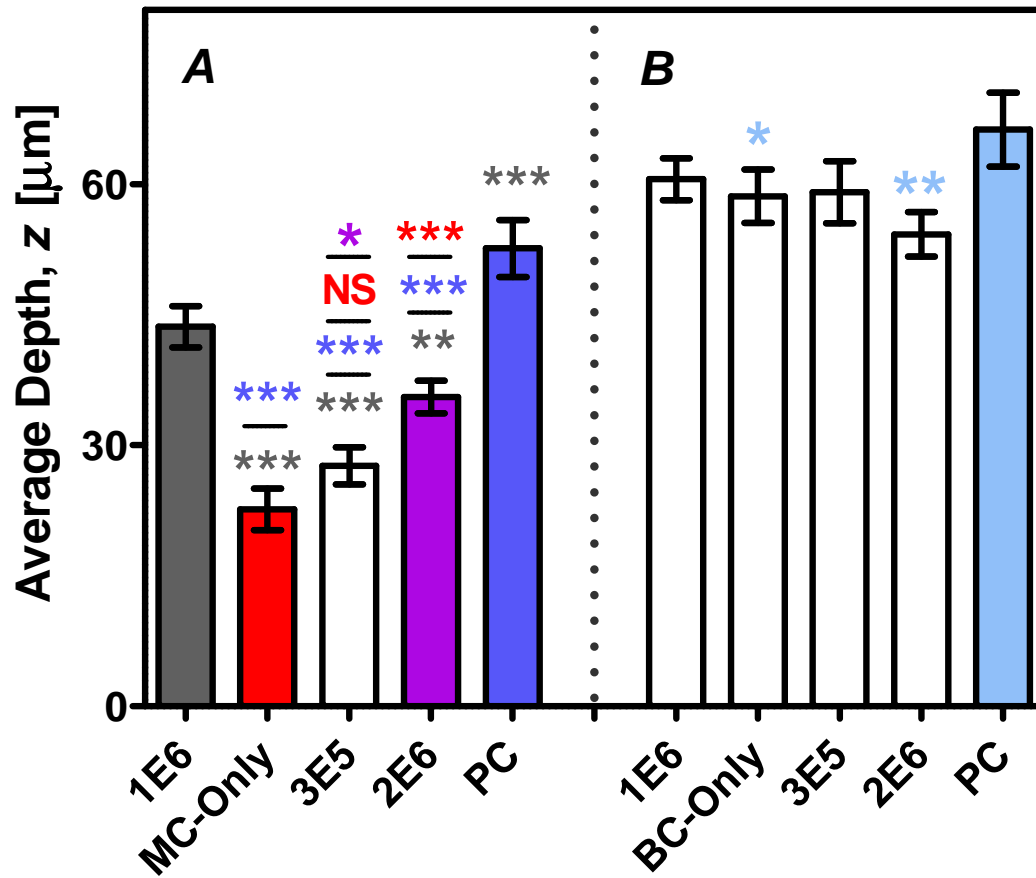


Figure 8. Impact of cell density in the bulk and presence of PCs on MC migration. (A) Intensity weighted average depth for monolayer cells were compared; (B) bulk cells were compared independently. All experiments were analyzed after 7 days of culture. The code name and sample relationship can be found in Table 1. Error bars: SD (generated by error propagation calculations from the depth distribution profiles). (NS: no significance; *: $p < 0.05$; **: $p < 0.01$; ***: $p < 0.001$)

respectively). We observe no significant differences in the penetration depths of MCs between MC-only case and 3×10^5 case (**Fig. 8A**, red and white, respectively). Interestingly, the 2×10^6 case showed significantly decreased penetration when compared to 1×10^6 case (**Fig. 8A**, purple and gray, respectively). We speculate that this is an effect of a diffused penetration due to increased branching of vessel structures: higher bulk density increases the network branches in all directions, making MCs less likely to move deeper by random walk. As an alternative hypothesis individual MCs may migrate from the monolayer into the bulk at day 1, prior to forming an angiogenic sprout, then form intermixed structures in both the bulk and at the interface without the requirement of an angiogenic invasion.

Pericytes (PCs) are known to associate with blood vessels and aid in their stabilization by providing quiescence cues to the emerging vasculature [25-26]. When PCs were introduced in the bulk of the collagen along with BCs (**Table 1-6**) we found that the distance of migration of MCs into the bulk was significantly higher relative to all cases with only ECs in the bulk (**Fig. 8A**, blue). In addition, the presence of PCs led to a significant difference in the distribution of BCs: less migration occurred toward the monolayer (**Fig. 8B**, blue). This observation suggests that PCs may have a stabilizing effect on the BCs as they were undergoing vasculogenesis. This stabilization is coherent with the proposition that MCs preferentially migrate on BCs due to a structural support (**Fig. 6C**, Day 7). This stabilization may result from the different protein composition that MCs experience given that the recruitment of PCs leads to basement membrane deposition [27] and their ability to release growth factors that induce angiogenesis [15].

3-3.4 Monolayer cells anastomose with bulk cells in a microfluidic setup

We exploited the anastomosis between ECs creating vascular networks within the bulk of a collagen matrix and ECs undergoing invasion angiogenesis in a microfluidic context (**Fig. 1B**). **Fig. 9A** shows a cell-seeded collagen device lined with GFP-HUVECs in one channel. We hypothesized that MCs would interact with BCs and create conduits for continuous flow from large engineering vessels to spontaneously created vascular tissue. After 7 days in culture with vasculogenesis media, luminal invasions were observed (**Fig. 9B,C**), as well as open lumen multi-cellular structures emerging in the bulk of the collagen (**Fig. 9A**). Two observations are important: (1) these luminal connections were made by the combination of MCs and BCs, and (2) these luminal spaces originated from the vessel wall of the EC-lined collagen channel (**Fig. 9 D-F**). We confirmed the open luminal structures by assessing the collagen fibers (**Fig. 9 G, H**). We note that these devices had a degree of leakage that created interstitial flows; these flows may have influence the viability and activity of the cells living in the bulk[28].

3-4 CONCLUSION

In our work we showed that angiogenesis and vasculogenesis can take place concurrently (**Fig. 5**). This concurrent activity of ECs led to a promising feature: the anastomosis of vascular structures inside a collagen matrix with angiogenic sprouts that originated from the monolayer that lined the surface of the collagen matrix. This phenomenon enabled continuous access from outside the culture setup into the bulk of the 3D culture (**Fig. 5C**). Importantly, this access could ease the design constraints in a microfluidic context: single large channels can interconnect with vessels in the bulk (**Fig. 9**) as opposed to designing complex 3D conduits that need to be lined

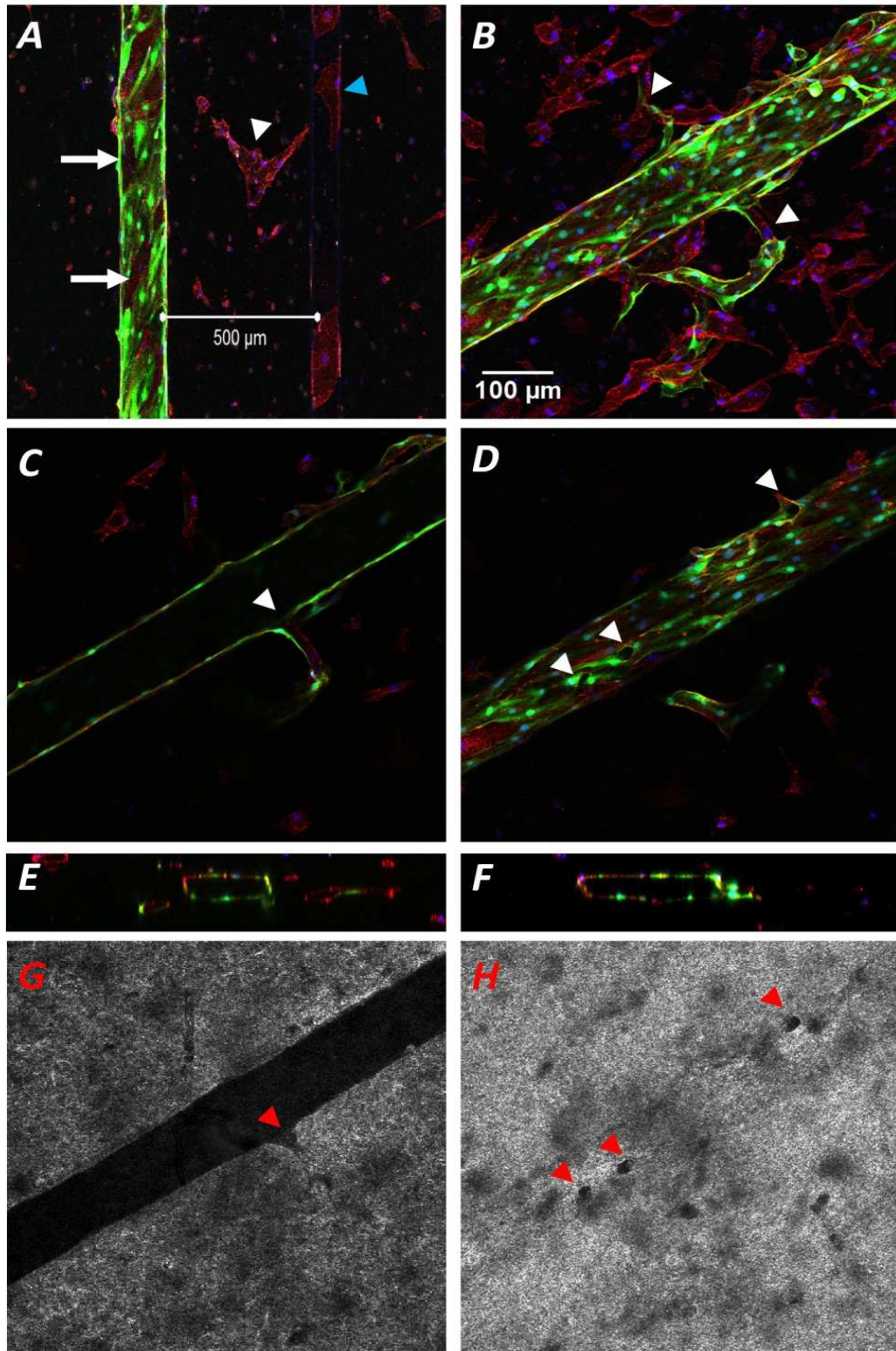


Figure 9. Confocal (**A-F**) and reflectance images (**G** and **H**) showing PiVLT culture quality and characteristics after 7 days. (**A**) The left channel is endothelialized with GFP-HUVECs while non-fluorescent HUVECs (red only, actin) are in the bulk of the collagen. After 7 days, cells from the bulk associate with the monolayer (white arrows) in a mosaic fashion, multicellular vacuolar structures emerge in the bulk (white arrowhead), and bulk HUVECs populate empty channel walls (blue arrowhead). The two channels are separated by a distance of 500 μm . (**B**) Az-projection through the channel height of an anastomosed endothelium. Bulk cells and monolayer cells interact to connect luminal spaces from the endothelialized channel into the bulk collagen (white arrows). Center slice (**C**, **E**) and bottom wall (**D**, **F**) of channel in **B** shows open-lumen emerging invading structures with a discontinuity in the endothelium wall (**C** and **D**, white arrowheads) and the void space found in a reflectance microscopy image to visualize collagen fibers (**D** and **F**, red arrowhead). zx-view (**E**) and zy-view (**F**) of z-stack (315 μm) for (**B**). Scale bar from (**B**) applies to (**C-D**), (**G-H**). In (**G**) and (**H**), other void spaces in the bulk collagen are due to vasculogenesis.

with ECs. The unexpected feature of the anastomosis process in our system was that the MCs and BCs populations did not connect with each other in a continuous manner (**Fig. 1Aii**). Instead, these populations were found intermixed to some extent and the intermixing was characterized by a migration of MCs into the bulk (**Fig. 6**).

We have identified a cellular interaction *in vitro* that could be analogous to the migration of circulating EPCs *in vivo*, where EPCs use the surface presented by blood vessel structures for their migration [29]. This process where cell networks are used as tracks (“cellular tracks”) has been described in several contexts as well. For instance, follicular dendritic cell networks served as a substratum for movement of follicular B cells within lymph nodes [30], angiogenic sprouts are found to be closely associated (or migrating) over astrocytic tracks [19], and motility of EC-associated PC depends on the presence of endothelial tubes [27]. In all the examples presented above, the interaction is between two different cell types. In our system, we identified the use of cellular tracks between cells of the same cell type. In **Chapter 2** we discussed how during vasculogenesis in 3D matrices individual cells do not communicate at a distance and angiogenic invasions were biased towards a boundary that served to support a contact mediated sprout, opposed to the traditional propositions that both mechanical and chemical cues may be signaling cells to interact. The results presented here along with the results in **Chapter 2**, suggest that there may be a generalized solution to the 3D migration of cells over large distances (larger than cell size) in tissues: cells simplify 3D migration into a 2D migration by using or creating ECM-free surfaces.

The different cellular conditions tested in this work and their different effects on anastomosis and the migration of MCs on BCs suggest a recipe to build vascular tissue over macroscopic distances to deal with mass transport at such scale (the engineering channels serve

as flow conduits from the outside world) and mass transport at the Krogh length scale given that a capillary bed is connected to the larger vessels (**Fig. 9**). This capacity was achieved by using local self-organization of vessel structures in 3D collagen matrices. The platform discussed here combined with the advantages of microfluidics (controlling the temporal and spatial distribution of solutes or physical forces) could advance the field of tissue engineering into its ambitious goals (**Fig. 10A**). For instance, tissues may be grown *in vitro* for implantation and the surgical anastomosis can be facilitated by the simple manipulation of the caliber of the feeder vessels. It may also serve as a system to study diseases *in vitro*, such as cancer [31]. Finally this platform may also serve to study the developmental stages of the vascular system at the tissue scale (**Fig. 10B-C**) in a controllable fashion [32-33].

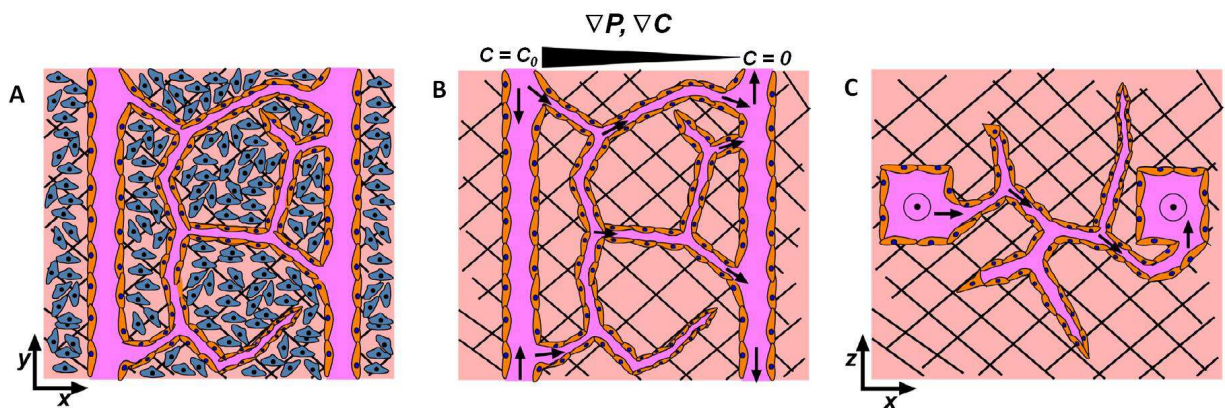


Figure 10. Future directions with PiVLT platform. (A) The identity of the bulk can be manipulated to culture different types of tissues that are nourished by the spontaneously formed blood vessels (specific cell type from tissue of interest in blue). (B) The two channels serve to connect outside flow into a fully percolated tissue after anastomosis (short black arrows). The outside flow can be used to tune the physical (pressure gradients, shear stresses) and chemical (soluble signal gradients) microenvironment of a maturing network of vessels. (C) Cross-sectional view of PiVLT shown in (A) and (B) depicts the three-dimensionality of the blood vessel networks.

3-5 REFERENCES

1. Ikada, Y., *Challenges in tissue engineering*. J R Soc Interface, 2006. **3**(10): p. 589-601.
2. Novosel, E.C., C. Kleinhans, and P.J. Kluger, *Vascularization is the key challenge in tissue engineering*. Advanced Drug Delivery Reviews, 2011. **63**(4-5): p. 300-311.
3. Gallico, G.G., 3rd, et al., *Permanent coverage of large burn wounds with autologous cultured human epithelium*. N Engl J Med, 1984. **311**(7): p. 448-51.
4. O'Connor, N.E., Mulliken, J.B., Banks-Schlegel, S., Kehinde, O., Green, H., *Grafting of burns with cultured epithelium prepared from autologous epidermal cells*. Lancet, 1981. **1**(8211): p. 75-8.
5. Atala, A., et al., *Tissue-engineered autologous bladders for patients needing cystoplasty*. Lancet, 2006. **367**(9518): p. 1241-6.
6. Shinoka, T., et al., *Creation of viable pulmonary artery autografts through tissue engineering*. J Thorac Cardiovasc Surg, 1998. **115**(3): p. 536-45; discussion 545-6.
7. Naito, Y., et al., *Successful clinical application of tissue-engineered graft for extracardiac Fontan operation*. J Thorac Cardiovasc Surg, 2003. **125**(2): p. 419-20.
8. Griffith, C.K. and S.C. George, *The Effect of Hypoxia on In Vitro Prevascularization of a Thick Soft Tissue*. Tissue Engineering Part A, 2009. **15**(9): p. 2423-2434.
9. Cross, V.L., et al., *Dense type I collagen matrices that support cellular remodeling and microfabrication for studies of tumor angiogenesis and vasculogenesis in vitro*. Biomaterials, 2010. **31**(33): p. 8596-8607.
10. Levenberg, S., et al., *Differentiation of human embryonic stem cells on three-dimensional polymer scaffolds*. Proc Natl Acad Sci U S A, 2003. **100**(22): p. 12741-6.
11. Choi, N.W., et al., *Microfluidic scaffolds for tissue engineering*. Nature Materials, 2007. **6**(11): p. 908-915.

12. Krogh, A., *The number and distribution of capillaries in muscles with calculations of the oxygen pressure head necessary for supplying the tissue*. J Physiol, 1919. **52**(6): p. 409-15.
13. Carmeliet, P. and R.K. Jain, *Angiogenesis in cancer and other diseases*. Nature, 2000. **407**(6801): p. 249-257.
14. Zheng, Y., et al., *Microstructured templates for directed growth and vascularization of soft tissue in vivo*. Biomaterials, 2011. **32**(23): p. 5391-5401.
15. Levenberg, S., et al., *Engineering vascularized skeletal muscle tissue*. Nature Biotechnology, 2005. **23**(7): p. 879-884.
16. Germain, L., M. Remy-Zolghadri, and F. Auger, *Tissue engineering of the vascular system: from capillaries to larger blood vessels*. Medical & Biological Engineering & Computing, 2000. **38**(2): p. 232-240.
17. Fuchs, S., A. Hofmann, and C.J. Kirkpatrick, *Microvessel-like structures from outgrowth endothelial cells from human peripheral blood in 2-dimensional and 3-dimensional co-cultures with osteoblastic lineage cells*. Tissue Engineering, 2007. **13**(10): p. 2577-2588.
18. Koike, N., et al., *Tissue engineering: creation of long-lasting blood vessels*. Nature, 2004. **428**(6979): p. 138-9.
19. Betsholtz, C., et al., *VEGF guides angiogenic sprouting utilizing endothelial tip cell filopodia*. Journal of Cell Biology, 2003. **161**(6): p. 1163-1177.
20. Song, J.W., D. Bazou, and L.L. Munn, *Anastomosis of endothelial sprouts forms new vessels in a tissue analogue of angiogenesis*. Integrative Biology, 2012. **4**(8): p. 857-862.
21. Morgan, J.P., et al., *Formation of microvascular networks in vitro*. Nat Protoc, 2013. **8**(9): p. 1820-36.
22. Bevington, P.R., Robinson D.K, ed. *Data Reduction and Error Analysis for the Physical Sciences*. 3rd ed. McGraw-Hill Higher Education. 2003, Kent A. Peterson: New York.
23. Davis, G.E., K.J. Bayless, and A. Mavila, *Molecular basis of endothelial cell morphogenesis in three-dimensional extracellular matrices*. Anatomical Record, 2002. **268**(3): p. 252-275.

24. Cross, V., *Biophysical and Chemical Mechanisms of Cellular Organization and Dynamics during 3-D Vasculogenesis*, in *Biomedical Engineering*. 2010 Cornell University: Ithaca.
25. Davis, G.E., et al., *Coregulation of vascular tube stabilization by endothelial cell TIMP-2 and pericyte TIMP-3*. *Journal of Cell Biology*, 2006. **175**(1): p. 179-191.
26. von Tell, D., A. Armulik, and C. Betsholtz, *Pericytes and vascular stability*. *Exp Cell Res*, 2006. **312**(5): p. 623-9.
27. Stratman, A.N., et al., *Endothelial-derived PDGF-BB and HB-EGF coordinately regulate pericyte recruitment during vasculogenic tube assembly and stabilization*. *Blood*, 2010. **116**(22): p. 4720-30.
28. Ng, C.P., C.L. Helm, and M.A. Swartz, *Interstitial flow differentially stimulates blood and lymphatic endothelial cell morphogenesis in vitro*. *Microvasc Res*, 2004. **68**(3): p. 258-64.
29. Gao, D., et al., *Endothelial progenitor cells control the angiogenic switch in mouse lung metastasis*. *Science*, 2008. **319**(5860): p. 195-8.
30. Bajenoff, M., et al., *Stromal cell networks regulate lymphocyte entry, migration, and territoriality in lymph nodes*. *Immunity*, 2006. **25**(6): p. 989-1001.
31. Stroock, A.D. and C. Fischbach, *Microfluidic culture models of tumor angiogenesis*. *Tissue Eng Part A*, 2010. **16**(7): p. 2143-6.
32. le Noble, F., et al., *Flow regulates arterial-venous differentiation in the chick embryo yolk sac*. *Development*, 2004. **131**(2): p. 361-75.
33. Lucitti, J.L., et al., *Vascular remodeling of the mouse yolk sac requires hemodynamic force*. *Development*, 2007. **134**(18): p. 3317-26.

CHAPTER 4

CONCLUSIONS AND FUTURE DIRECTIONS

In Chapter 1 we discussed the state of the art for the fundamental question of cell-cell communication during vascular morphogenesis. Specifically, we outlined the considerations for self-organization, overviewed the molecular biology of vascular development, discussed the schools of thought for fundamental mechanisms that lead to cell-cell communication and identified the limitations of the art. In addition, we highlighted the importance of flow and physical forces during the development of the vascular system during angiogenesis. We further identified the limitations of current tissue engineering systems; the biggest being proper nourishment of developing tissues *in vitro* [1]. Multi- and heterotypic cellular systems such as tissues require organization at a length scale of many cells, highlighting the importance of cell-cell communication. The study of cell-cell communication during the development of vascular structures became the main focus of the thesis.

In Chapter 2, we studied the behavior of individual and multi-cellular processes during vasculogenesis and angiogenesis, respectively. In the context of sprouting angiogenesis within 3D collagen scaffolds, we observed that cells responded to the presence of solid boundaries to which the collagen matrix was bound. We further observed that the preponderance of invasions near boundaries was due to their direct association with the boundaries. This behavior is compatible with a mechanism of cell-microenvironment interaction based on contact. We point out that this contact-based response of sprouting angiogenesis was in the absence of externally applied gradients of soluble factors. In both *in vivo* [2] and *in vitro* [3], angiogenic invasions respond to gradients of VEGF by aligning the invasions with the direction of the gradient. In

order to further understand the nature of the contact-mediated response of invading sprouts, we present recommendations for future work. **Figure 1** shows two cells extending protrusions towards each other. ECs have polymerized a fibronectin (FN) matrix around them (shown in green) that extends beyond the cell cytoskeleton (red, actin). FN polymerization involves opening of cryptic sites for attachment of cell adhesion sites and for the interaction of other extracellular matrix (ECM) proteins [4-5]. The cell-driven FN polymerization is crucial for vascular morphogenesis as it allows the cells to rearrange and engage with the surrounding ECM [6]: if FN polymerization is inhibited, invading tubules and vasculogenesis are also inhibited [7]. In our invasion angiogenesis assay, we could test the following hypothesis: FN polymerization of cells fully embedded in compliant collagen is not as efficient as FN polymerization of cells near a solid boundary. Near a solid boundary, secreted FN would attach not only to the ECM but the boundary as well. The binding to the boundary provides an unmovable surface with which the cells can exert more tension, compared to anchoring onto movable collagen fibers in fully compliant regions of the matrix. In **Fig. 1**, the serum in the media was depleted of FN; the FN is FITC-labeled and added to the media (FITC labeling of FN is described [8]). However, in future experiments a better representation of what cells are doing with secreted FN can be achieved by transducing the cells with GFP-FN [9]. Image analysis could be performed on samples in order to quantify FN near boundaries and far from boundaries. A complimentary experiment would involve treating the boundary surfaces with the FN polymerization inhibitors [7] as a negative control.

In the context of vasculogenesis, we observed in Chapter 2 that individual cells required a minimum separation to establish connection in 3D collagen scaffolds. In addition, ECs did not respond to neighboring cells by altered behavior such as migration or directionality of protrusive

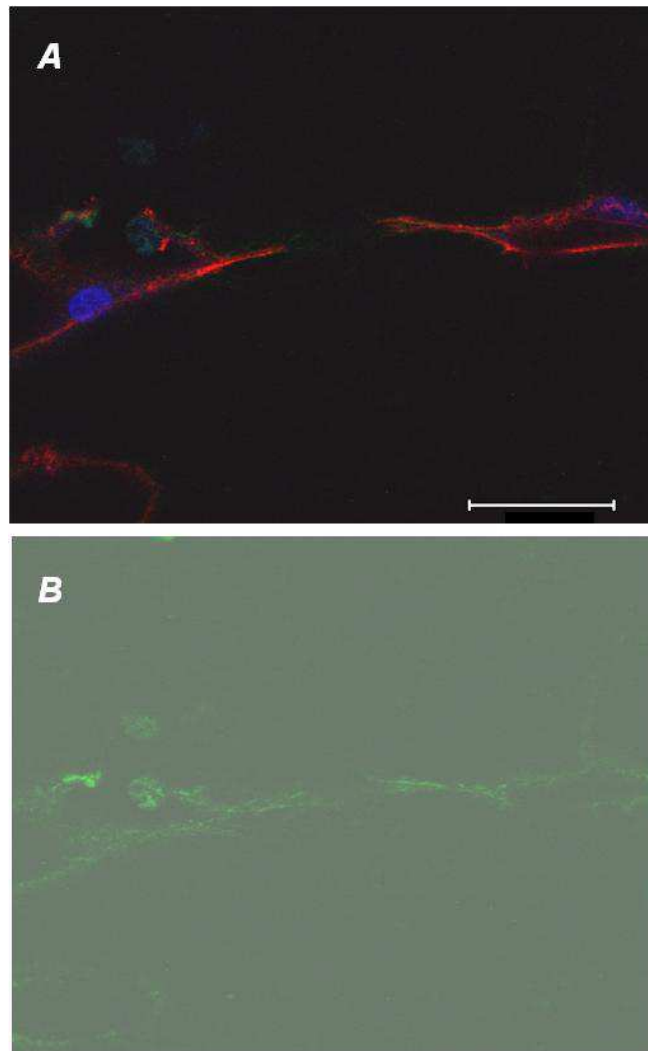


Figure 1. FN polymerization during vascular morphogenesis. Confocal micrograph (40x objective) of two cells extending protrusions towards each other. **(A)** FN (green), actin (red), cell nucleus (blue, DAPI). **(B)** Brightfield image superimposed with FN (green); presented for clarity of the FN.

activity. These observations suggest that communication at a distance is not taking place during vasculogenesis in 3D environments. However, alteration of migration patterns [10-11] and directionality of protrusive activity in response to gradients of VEGF [12] have been reported for ECs on 2D substrates. As discussed in Chapter 2, we do not discard the possibility that mechanical signals played a role in the observed contact-dependent responses of ECs. Due to the intricate 3D network of fibers, the decay of the tractional field from a cell could be very short, and beneath the resolution of our experiments. In the potential use of mechanical communication, ECs would have to overcome the limitation of fast decay in tractional field by secreting additional ECM proteins or by bundling the surrounding collagen [13]. The local increase in fiber concentration transduces traction faster at the expense of distance [14]. Further understanding of the interaction between individual cells and the matrix could be achieved by studying both in a concurrent manner, in particular for cells in pairs ($< 40 \mu\text{m}$ initial separation). **Figure C-5** in Appendix C shows an individual cell constitutively expressing GFP and the surrounding collagen fibers (in red) revealed by reflectance microscopy. This capacity can be exploited with the implementation of live imaging to track the dynamics of cells along with fibers. In addition, our experimental results suggest that pre-positioning of individual cells may play an important role in the capacity of ECs to establish connections and form a network. In a system where individual cells are pre-positioned in a 3D point to form specific geometrical shapes, a set of biophysical considerations could be addressed: (1) whether the set of cells connect into a vascular structure taking on the shape of the geometry they were pre-positioned to (this could imply an active use of protrusive activity for connectivity); (2) whether the set of cells always connect into the typical circular cross-sectional structure regardless of their pre-positioning shape (this could imply that cells directly attach using an enveloping mechanism);

(3) whether clusters of cells in any pre-positioned geometry are required to establish a long-range (> 40 μm , from the requirement for cell-cell connectivity in Chapter 2) connection between two clusters in the form of a tube. Pre-positioning of single cells to address the above considerations may be achieved using the newly developed Block-Cell-Printing technique by Zhang, et. al. [15]. In brief, this technique employs microfluidic arrays of hook-shaped traps to hold cells at designated positions and directly transfer the anchored cells onto various substrates. For our experiments, the substrates could be a collagen gel that could later be sandwiched by another collagen layer. Such an environment could allow for pre-positioning of individual cells in a 3D environment that allows for similar experiments discussed in Chapter 2 to be performed. A better understanding of cell-cell communication during vasculogenesis would not only expand fundamental knowledge in the vascular biology community, but could also enable the engineering of perfusable tissues, as we presented in Chapter 3.

In Chapter 3, we hypothesized that the combination of the vasculogenesis and angiogenesis processes generated connected structures that were characterized by a continuous luminal space from angiogenic vessels into network vessels generated by vasculogenesis; this represents a hierarchical vascular structure *in vitro*. We showed that anastomosis could be achieved using this strategy. This approach could lead to a significant simplification of microfluidic designs that make extensive use of vessel lengths, bifurcations and branching in order to achieve a level of nourishment that enables tissue survival [16-17]. Some challenges remain to be addressed for our approach: (1) the optimization of the seeding of MCs onto the channel surfaces for uniform coverage along the channel length, (2) the assessment of the impact of device leakage mentioned in Chapter 3 on device performance for tissue survival (this leakage represents an interstitial flow through the collagen matrix), (3) the identification of an optimal

separation between different feeder channels for bigger constructs, and (4) the establishment of a method of continuous perfusion of the system. Nonetheless, this platform represents a unique tool that could open the possibility for the growth of thick tissues *in vitro*, and to study the maturation of the vascular system. For instance, using our combined angiogenesis and vasculogenesis approach that results in our Perfuseable in Vitro Living Tissue (PiVLT), a flow system can be attached in the microfluidic setting which could prevent a living tissue from undergoing necrosis since it is being nourished by media in the bulk. **Figure 2A** shows our PiVLT system used to grow a tissue of a specific type (e.g. cancer cells [18]). In the context of vascular maturation, the PiVLT can be exploited to study a range of events that arise during *in vivo* maturation: the role of physical forces on network architecture [19] and the impact of network location and vessel caliber on pericyte association and identity [20]. Physical forces (e.g. shear stress) are a requirement for vessel remodeling in the developing vasculature [19]. Using the PiVLT, we could further understand this role on network architecture in environments of controlled pressure gradients. The magnitude of pressure gradients that a developing vasculature experiences could correlate to a characteristic length at which the architecture transitions from large caliber vessels into capillary-sized vessels (**Fig. 2B-D**). Combining this capacity with microscopy, we could also exploit our PiVLT system in time-lapse experiments to track changes that take place during network pruning (e.g. vessel regressions and dilations). In a hierarchical structure generated in our PiVLT, a next step in the maturation process could also be studied: perivascular cell association. Pericytes are recruited to the abluminal surface vessels to promote stability. It has been observed that capillaries are covered with pericytes while large vessels (e.g. arteries, veins, arterioles, venules) are covered by smooth muscle cells (SMCs) that arise from the same cell lineage [20]. Using our PiVLT, a hypothesis that the nature of the

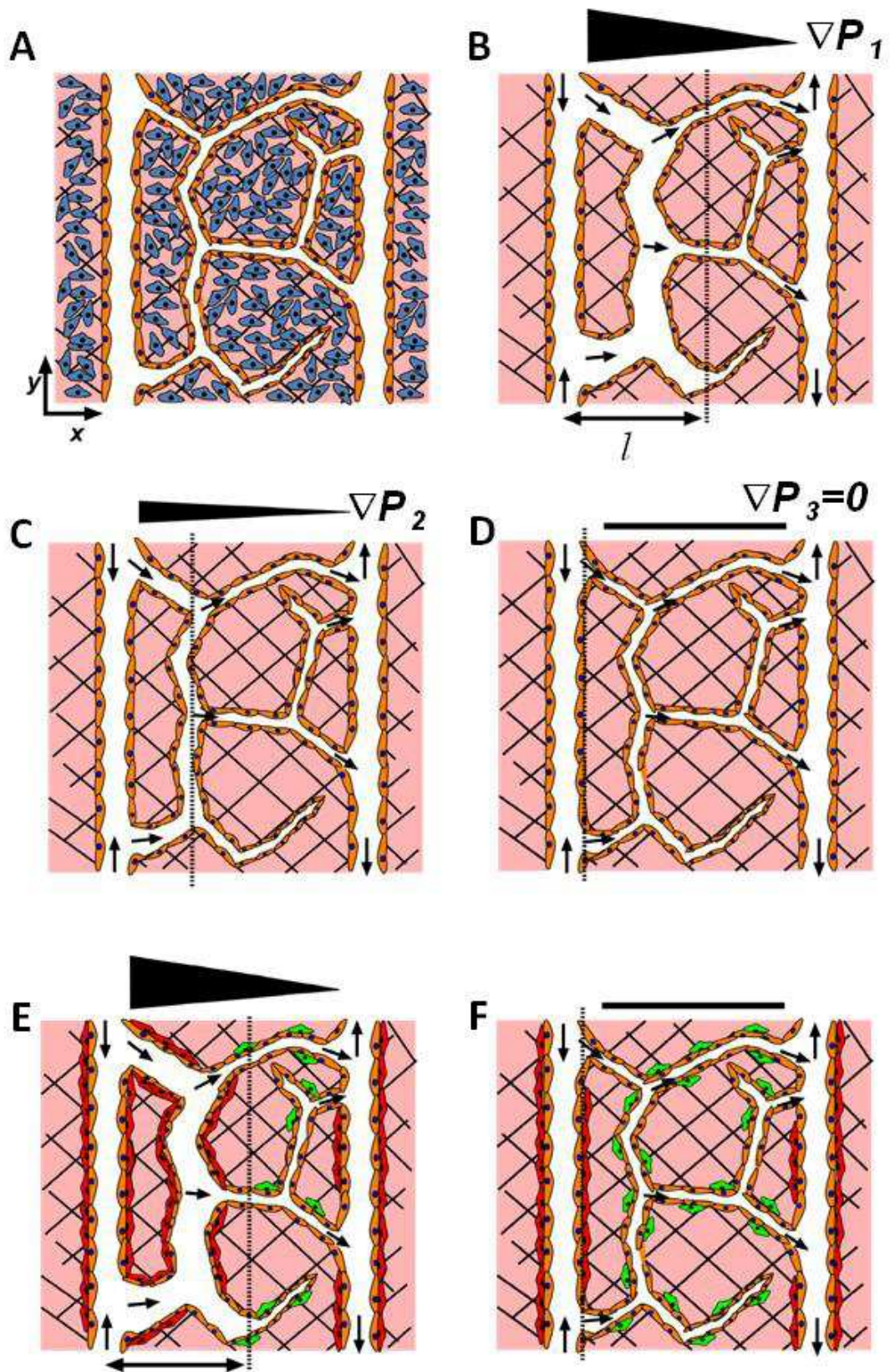


Figure 2. Hypothetical study of physical forces and perivascular cell associations using PiVLT. (A) The PiVLT can be used to grow tissues, such as a model tumor *in vitro*. Blue cells represent cancer cells seeded in the bulk of the collagen. Vascular structure emerged from the interaction of cells undergoing invasion angiogenesis and cells undergoing vasculogenesis, concurrently. (B-D) PiVLT is exposed to different gradients of pressure to study the role of physical forces on the architecture of the network; $\nabla P_1 > \nabla P_2 > \nabla P_3$. (B-C) The vasculature inside the PiVLT responds to the pressure gradient by dilating the vessels in proximal to the high pressure side. The extent of the response is a characteristic length, l , that describes the distance over which network dilation occurs; $l_1 > l_2$. (D) For no pressure gradient; $l = 0$. (E-F) Hypothetical framework on perivascular cell associations as a function of vessel caliber. (E) A high pressure gradient establishes a hierarchical structure as in (B). When perivascular cells are in the bulk of the culture they are recruited to the vessel walls. Large vessels can differentiate perivascular cells into Smooth Muscle Cells (red cells) while small vessels are sparsely covered by pericytes (green cells). (F) No pressure gradient would exhibit a uniform distribution of pericytes that sparsely cover capillary vessels.

perivascular cells is dependent on vessel caliber could be tested (**Fig. 2E-F**). In such an experiment, the association of the perivascular cells could exhibit different protein expression based on whether they are SMCs or remain as pericytes. With all these potential experiments we could generate a set of design rules in order to create thick tissues using a bottom-up approach.

Both the field of vascular biology and the field of tissue engineering may benefit from the work that has been discussed here. We presented a new perspective for cell-cell communication and migration in 3D environments during vascular development, which are not consistent with current views for cell-cell communication and migration on 2D substrates. These inconsistencies may be a result of the difference in microenvironmental conditions, such as an additional dimension for external mechanobiological signals or for cell adhesion, and the different diffusive profiles of solutes around cells [21]. In addition, we presented a unique approach to solve a central tissue engineering challenge: the anastomosis of angiogenic sprouts with vascular networks originating from vasculogenesis. This approach may represent a significant step in the tissue engineering community given that a major limitation of the field is tissue nourishment [1]. In order to move toward creating tissues that are implantable, this anastomosis approach should be exploited: it presents simple designs that have ends for surgical attachments. Further work (as discussed above) will improve the design considerations for the PiVLT that lead to successful maintenance of healthy thick tissues *in vitro*. The ambitious goal that starts with the use of the PiVLT is to grow replaceable tissue sections and even to create full organs in the future.

4-1 REFERENCES

1. Novosel, E.C., C. Kleinhaus, and P.J. Kluger, *Vascularization is the key challenge in tissue engineering*. Advanced Drug Delivery Reviews, 2011. **63**(4-5): p. 300-311.
2. Betsholtz, C., et al., *VEGF guides angiogenic sprouting utilizing endothelial tip cell filopodia*. Journal of Cell Biology, 2003. **161**(6): p. 1163-1177.
3. Chen, R.R., et al., *Integrated approach to designing growth factor delivery systems*. FASEB J, 2007. **21**(14): p. 3896-903.
4. Klotzsch, E., et al., *Fibronectin forms the most extensible biological fibers displaying switchable force-exposed cryptic binding sites*. Proc Natl Acad Sci U S A, 2009. **106**(43): p. 18267-72.
5. Mosher, D.F. and P.E. Schad, *Cross-linking of fibronectin to collagen by blood coagulation Factor XIIIa*. J Clin Invest, 1979. **64**(3): p. 781-7.
6. Sottile, J. and D.C. Hocking, *Fibronectin polymerization regulates the composition and stability of extracellular matrix fibrils and cell-matrix adhesions*. Mol Biol Cell, 2002. **13**(10): p. 3546-59.
7. Zhou, X., et al., *Fibronectin fibrillogenesis regulates three-dimensional neovessel formation*. Genes Dev, 2008. **22**(9): p. 1231-43.
8. McKeown-Longo, P.J. and D.F. Mosher, *Interaction of the 70,000-mol-wt amino-terminal fragment of fibronectin with the matrix-assembly receptor of fibroblasts*. J Cell Biol, 1985. **100**(2): p. 364-74.
9. Ohashi, T., D.P. Kiehart, and H.P. Erickson, *Dynamics and elasticity of the fibronectin matrix in living cell culture visualized by fibronectin-green fluorescent protein*. Proc Natl Acad Sci U S A, 1999. **96**(5): p. 2153-8.
10. Reinhart-King, C.A., M. Dembo, and D.A. Hammer, *Cell-Cell Mechanical Communication through Compliant Substrates*. Biophysical Journal, 2008. **95**(12): p. 6044-6051.
11. Yin, Z.Z., et al., *Analysis of pairwise cell interactions using an integrated dielectrophoretic-microfluidic system*. Molecular Systems Biology, 2008. **4**.

12. Shamloo, A., et al., *Endothelial cell polarization and chemotaxis in a microfluidic device*. Lab on a Chip, 2008. **8**(8): p. 1292-1299.
13. Cross, V.L., et al., *Dense type I collagen matrices that support cellular remodeling and microfabrication for studies of tumor angiogenesis and vasculogenesis in vitro*. Biomaterials, 2010. **31**(33): p. 8596-8607.
14. Califano, J.P. and C.A. Reinhart-King, *A Balance of Substrate Mechanics and Matrix Chemistry Regulates Endothelial Cell Network Assembly*. Cellular and Molecular Bioengineering, 2008. **1**(2-3): p. 122-132.
15. Zhang, K., et al., *Block-Cell-Printing for live single-cell printing*. Proc Natl Acad Sci U S A, 2014. **111**(8): p. 2948-53.
16. Miller, J.S., et al., *Rapid casting of patterned vascular networks for perfusable engineered three-dimensional tissues*. Nat Mater, 2012. **11**(9): p. 768-74.
17. Kang, T.Y., et al., *Design and assessment of a microfluidic network system for oxygen transport in engineered tissue*. Langmuir, 2013. **29**(2): p. 701-9.
18. Stroock, A.D. and C. Fischbach, *Microfluidic culture models of tumor angiogenesis*. Tissue Eng Part A, 2010. **16**(7): p. 2143-6.
19. Lucitti, J.L., et al., *Vascular remodeling of the mouse yolk sac requires hemodynamic force*. Development, 2007. **134**(18): p. 3317-26.
20. Hirschi, K.K. and P.A. D'Amore, *Pericytes in the microvasculature*. Cardiovasc Res, 1996. **32**(4): p. 687-98.
21. Griffith, L.G. and M.A. Swartz, *Capturing complex 3D tissue physiology in vitro*. Nat Rev Mol Cell Biol, 2006. **7**(3): p. 211-24.

APPENDIX A

EXPERIMENTAL PROTOCOLS

A-1. Passaging human umbilical vein endothelial cells (HUVECs) and human brain vascular pericytes (HBVPs)

HUVECs

1. One flask of 80-90% confluent cells splits into 4 or 5 new flasks.
2. Warm up Trypsin, HEPES (or PBS), and Growth Media (GM) to 37°C.
3. Decant cell media inside the culture flask containing the cells to passage into a waste beaker. Special consideration in avoiding contact between flask opening and beaker should be taken into account.
4. Rinse once with warm HEPES. Typical volume of HEPES used for a T75 flask (75 cm²) is 3 mL. For a T25 (25 cm²), 1 mL is sufficient.
5. Decant HEPES volume into waste beaker. Special consideration in avoiding contact between flask opening and beaker should be taken into account.
6. Add warm Trypsin to the flask. Typical volume for a T75 flask is 3 mL.
7. Expose the cells to Trypsin for 30 seconds.
8. Discard Trypsin into waste beaker.
9. Tap the side of the flask to lift-off cells.
10. Add GM to the bottom wall of the flask. Transfer split volumes of the suspension into new flasks containing warm GM (10 mL).

HBVPs

1. Same as for HUVECs but wash two times with HEPES and instead of using GM, use Pericyte Media (PM) that is prepared according to manufacturer.

A-2. Preparation of Growth Media (GM) and Vasculogenesis Media (VM)

Growth Media (GM)

1. Take one 500 mL bottle of M199 and mix with the following:
 - a. 7.5 mL of stock Pen/Strep (for a working concentration of 150 Units/mL)
 - b. 6 mL of stock L-glutamine (for a 2 mM working concentration)
 - c. 250 µL of a 10,000 Units/mL stock solution of heparin (for a working concentration of 5 Units/mL)
 - d. 100 mL of stock FBS (for a working concentration of 18%)
 - e. 15 mg (one vial) of ECGS (for a working concentration of 20 µg/mL)
2. Cap and mix gently by inverting slowly without creating too many bubbles (the FBS enhances bubble formation).
3. Connect a filtration unit (filter/bottle system) and attach it to a vacuum pump.
4. Sterile filter the solution.

Vasculogenesis Media (VM)

1. Determine the volume of VM needed (typically VM is only prepared as needed).
2. Calculate the volumes to be added into the solvent. GM is used as the base media for VM. In this context, GM is the solvent. The following are to be added to the GM:
 - a. TPA at 50 ng/mL

- b. VEGF165 at 40 ng/mL (keep on ice while handling)
 - c. bFGF at 40 ng/mL (keep on ice while handling)
 - d. Ascorbic acid at 50 µg/mL
3. The difference between the added volumes of these substances and the final VM volume desired should be the volume to add of GM. In example, if 1mL of VM was needed, 2 µL of VEGF (from a 20 µg/mL stock), 1.6 µL of bFGF (from a 25 µg/mL stock), 1.6 µL of TPA (from a 50 µM stock), and 10 µL of Ascorbic Acid (from a 5,000 µg/mL stock) were to be added to 985 µL of GM.
 4. Warm up (or pre-warm GM). VM should be ready to use.

A-3. Immunohistochemistry

Here I describe the methodology used to perform staining on both, culture wells and microfluidic devices, which contain collagen and cells. Cells could be HUVECs, for which the typical specific markers were CD31; they could also be HBVPS, for which typical specific markers were Desmin or smooth muscle actin (SMA). Staining for filamentous actin and nuclei was generic for all cell types.

1. Remove cell media from the top of the gels by pipetting or aspirating with a Pasteur pipette attached to a vacuum pump.
2. Fix cells/gels with 3.7% formaldehyde in PBS for 30 mins.
3. Remove formaldehyde with pipette and place in formaldehyde waste tube.
4. Wash three times for 5 mins with PBS.

5. Permeabilize cell membranes and block for non-specific protein binding with 1% Triton-X/3% BSA in PBS for 1 hr.
6. Aspirate Triton X/BSA and place in waste tube.
7. For cell specific primary antibodies (1°Ab's) (such as CD31, Desmin, and SMA), expose the samples to a 1:50 (antibody:solvent) mixture of the intended antibody in 1% BSA in PBS overnight at 4°C. Determine an appropriate volume that will allow the entire sample to be bathed in the solution.
8. Remove antibody mixture using aspiration.
9. Wash with PBS three times for 5 mins each.
10. Secondary antibody (2°Ab) and generic stain mixtures should be prepared **in dark** to prevent degradation of the fluorophore (2°Ab's have the fluorophores attached to them). Determine an appropriate volume that will allow the entire sample to be bathed in the solution. Using 1% BSA in PBS, prepare a 1:100 solution of the generic and 2°Ab's:
 - a. Typical 1°Ab for CD31 used the laboratory is raised in rabbits against humans. It follows that a 2°Ab raised against rabbit is required.
 - b. Typical 1°Ab's for Desmin and SMA used the laboratory are raised in mice against humans. It follows that a 2°Ab raised against mouse is required.
 - c. Alexa Fluor Phalloidin (filamentous actin stain)
 - d. 5uM DAPI (from 500 µM)
 - e. These can be together in the same buffer solution.
 - f. As an example, a sample has HUVECs only and both 2°Ab and generic stains are desired. If 1 mL of generic and 2°Ab solution was needed, 10 µL of 2°Ab against

rabbit, 10 μ L of Phalloidin, and 10 μ L of DAPI were to be added to 970 μ L of 1% BSA in PBS.

11. Incubate for 1 hr at room temperature. Keep covered with Aluminum foil from now on.
12. Remove stain using aspiration.
13. Wash three times for 5 mins (minimum) with PBS.
14. Image samples ASAP.

A-4. Vasculogenesis well culture fabrication

1. A glass disc with a center through hole (drilling aided by Mr. Glenn Swan) is cleaned with soap and water, then dried. See **Figure A-1** for an illustration of the setup.
2. One gram of uncured mixed and degassed, PDMS is place on a cleaned cover slip.
3. Allow the liquid PDMS to spread on the entire surface and partially cure the PDMS in oven at 60°C for 25 mins. After this partial curing, the PDMS should have a tacky consistency.
4. Place the glass disc onto the partially cured PDMS and finish curing for 2 hrs at 60°C. The tacky consistency of partially cured PDMS will prevent the PDMS from crawling up the well and occupying the well space that would otherwise be cell/collagen occupied.
5. Punch holes of an appropriate size on a flat, thick piece of PDMS.
6. Drop a small amount of uncured PDMS on the surfaces of the remaining PDMS around the punched hole and bond it to the top part of the glass disc (that is not

bonded to the PDMS on the glass cover slip). This will serve as a reservoir for media to sit on top of the culture.

7. Cure for 2 hrs.
8. Expose the assembled device to O₂ plasma for 4 mins in order to sterilize. The culture well should be ready to use.

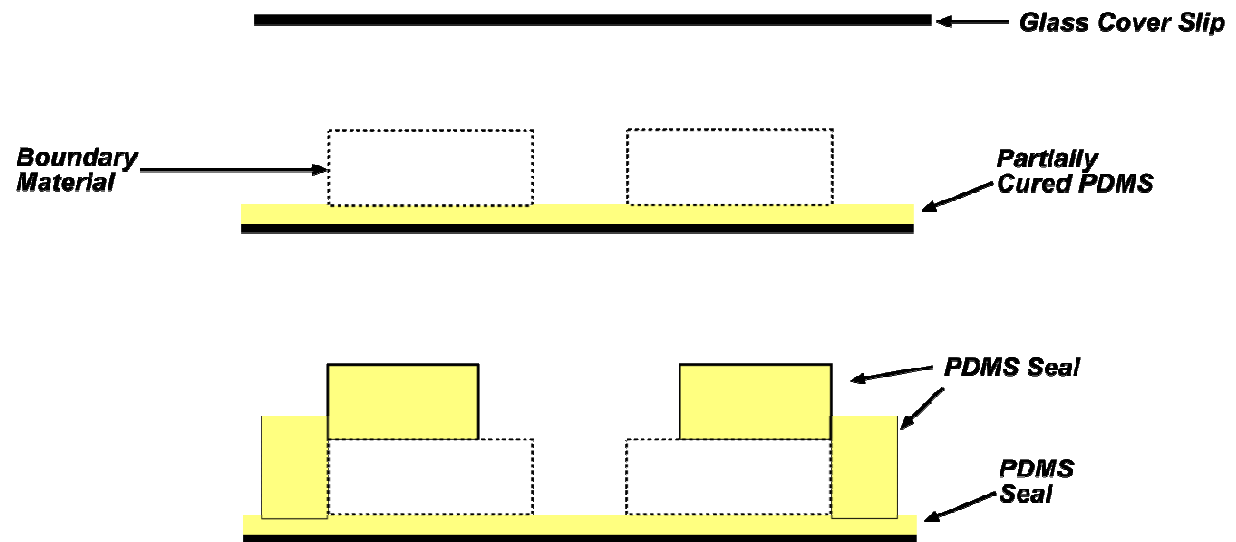


Figure A1. Vasculogenesis well culture fabrication steps. A thin layer of pre-cured PDMS is applied onto a glass cover slip. The PDMS is partially cured then the boundary material is pressed onto it. The entire assembly is later fully cured.

A-5. Invasion angiogenesis well culture fabrication

1. Glass discs were drilled (aided by Mr. Glenn Swan) partially (no through holes) with diameters of 4 mm and 1 mm. For an illustration of the discs see **Figure A2**.
2. Glass discs were cleaned with soap, rinsed and dried.
3. Expose the glass discs to O₂ plasma for 4 mins to sterilize. Ready to use.

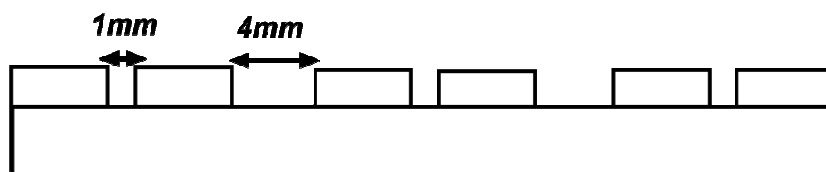


Figure A2. Glass discs with partially drilled holes.

A-6. Counting protrusions of individual cells

Here I describe the systematic approach and the rules that I used to consider protrusions for the analysis of protrusion dynamics of individual cells (Chapter 2 and Appendix).

Loading Sequence of Images Using the ImageJ freeware:

1. Select “File -> Import -> Image Sequence...”, and load the sequence of images that is to be analyzed.
2. To set scale, “Analyze -> Set Scale...”. In the “Distance in Pixels” box, type “3.2”. “Known Distance” box, type “1”. “Unit of Length” box, type “um”, then check the “Global” box. Press “OK” to continue. The conversion between distance in pixel and known physical distance is determined by the objective lens used; in this case, the magnification was 40x (Olympus LUC PPlanFL N 40x/0.6).
3. ImageJ needs to be told what to measure. For this we select “Analyze -> Set Measurements...”. Check the box for “Center of Mass” and “Invert Y Coordinates”. Press “OK” to continue. The inversion of the y-coordinates will set the origin for this coordinate to the bottom of the image. By default, this origin is at the top of the image.
4. The “Center of Mass” feature was used to manually identify the center of the cell (i.e. the center of the nucleus in **Figure A3**), the reference location and the location of a protrusion. To enable the acquisition using this feature, click on the “Point Selections” button. Double click on it to check the “Auto-Measure” box.
5. If tracking cell centers or reference locations, double click on the “Point Selections” option and also check the “Auto-Next Slice” box. This will allow for a continuous

tracking of the point of interest. When tracking protrusions, it is convenient not to have the “Auto-Next Slice” feature enabled. An individual protrusion is not trivially identified as it can be highly dynamic.

Rules to count protrusions:

6. An individual protrusion was recorded at its maximum length of existence (**Figure A3**). The existence of a protrusion was identified; scrolling back and forth in the sequence of images allowed to specify the time at which a protrusion reached its maximum length.
7. If a protrusion branched, two protrusions were counted based on the magnitude of separation of the daughter branches. For instance, if two daughter branches were examining two opposing directions, two protrusions were counted.
8. Some protrusions examined the same direction more than once. A protrusion in the same direction was counted as many times as it existed only if the previous one had retracted fully (and not partially) into the cell membrane.
9. If a protrusion extended beyond the field of view, its maximum extended location was the same as the location of field of view edge where it “escaped”.

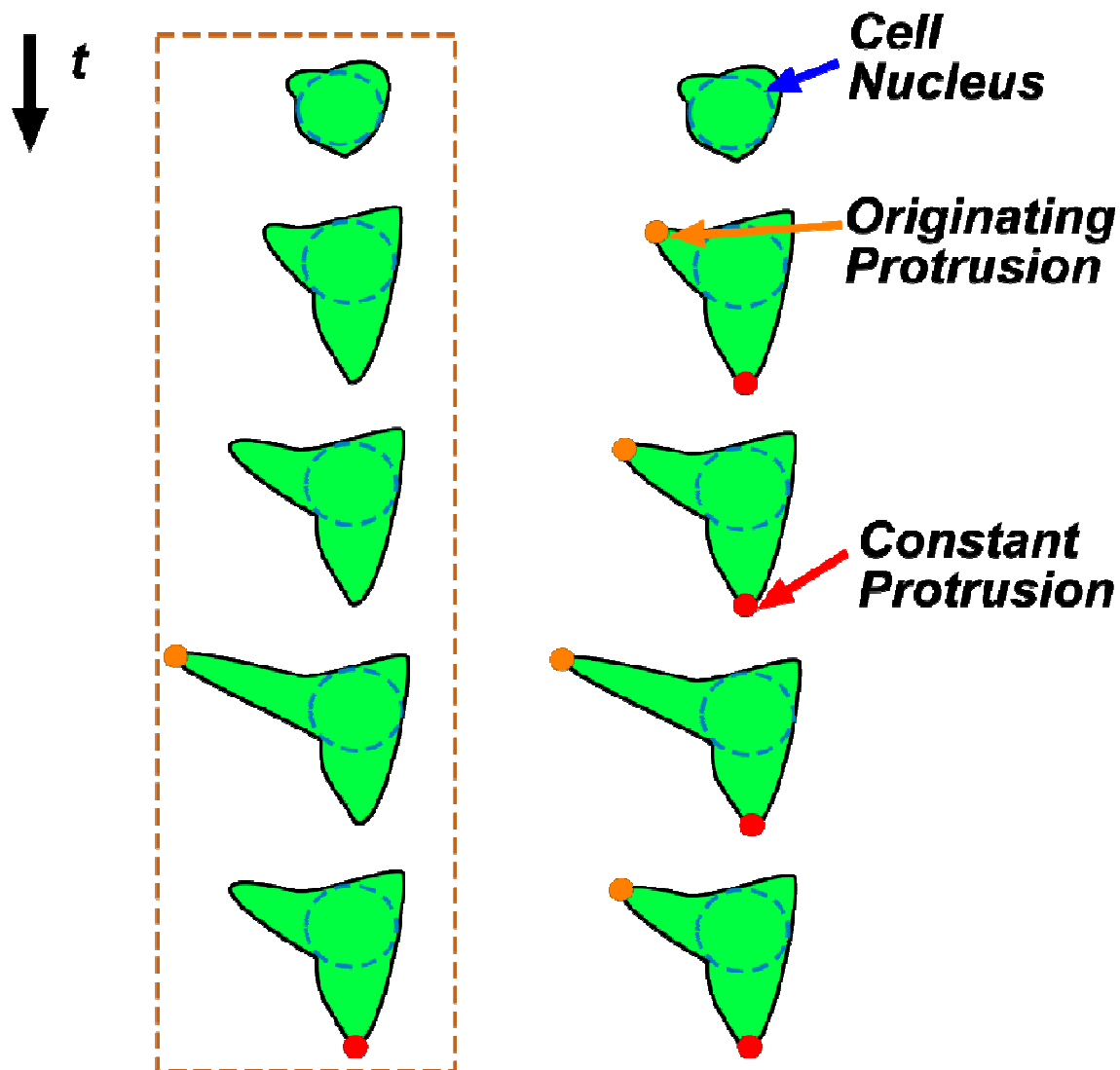


Figure A3. Selecting a protrusion. A protrusion can be sent initially and remain the same (“constant protrusion”) over time, while others are more dynamic. The center of mass for the cell structure was selected to be the center of the cell body. The cell body structure contains the nucleus, and in time, it also contains lumenized spaces.

A-7. Live imaging time-lapse setup

The *xy*-coordinates of individual cell protrusions were obtained from a sequence of images using ImageJ (see section A.6). The sequence of images was generated by continuously taking a picture of an individual cell of interest. Here I described how to setup the time-lapse imaging using the SlideBook™ software after the sample is in place and a cell of interest has been identified.

1. Press the “Focus” button on the software interface. This button has a distinctive “F” as the icon image.
2. Bring the cell center to focus using the scope dial.
3. On the “Focus Controls” panel, press the “XY” tab and press “Set Point”. This will save the *xyz*-position of the cell in reference to the microscope frame.
4. Find as many cells as desired but no more than 35 positions. The restriction originates from the limitation of manual refocusing. You need time to refocus every two hours without interrupting the time step predetermined for every image acquisition. To track these positions simply click on “Set Point” for every position of interest.
5. Set the binning to 2x2 to prevent saving excessive amounts of data as images.
6. Back in the SlideBook main screen, click on “Capture” icon. This icon has a distinctive camera cartoon on it.
7. Check the “3D” box, set the Step size to 25 μm and the number of planes to 3.
8. In the Timelapse settings, the interval is set to 10 mins and the estimated duration to 12 hrs (since the microscope has to be refocused manually every two hrs this can be any time).

9. Filter set square, click on TRANS.
10. Advanced-> Spool->Capture to spool file after each time point only-> create folder.

A-8. Perfusable in Vitro Living Tissue Assembly

1. The device is assembled using three major pieces: (1) top plexi-glass contains ports for gel loading and for flow through featured channels (most likely also contains built-in reservoirs); (2) bottom plexi-glass serves as support for sealing and has a slot that fits a cover slip; (3) 25mm x 25mm cover-slip that holds a collagen bottom layer. Make sure to also have at hand, sterilized short screws, PDMS slab, and Dowell pins. See **Figure A4**.
2. Activate surfaces of top plexi-glass piece and cover-slip
 - a. Expose surfaces to O₂ plasma for 4 minutes
 - b. Coat with 1% PEI for 10 mins
 - c. Rinse with sterile dH₂O
 - d. Coat with 0.1% GA for 30 mins
 - e. Rinse thoroughly with sterile dH₂O
 - f. Dry
3. Prepare cells to be added to the bulk of the collagen.
 - a. If more than needed cells are available, take out the amount of cells needed and spin down to minimize volume added later to the collagen mixture.
4. Prepare collagen/cell mixture
 - a. 0.6% collagen
 - b. Add cells needed (typical working density of 1x10⁶ cells/mL)
5. Place the activated 25mm x 25mm cover-slip in the slot of the plexi-glass assembly part.

6. Sterilize and increase wettability of PDMS stamp and slab by exposing it to O₂ plasma for 4 mins
 - a. Make use of tall petri dish
7. Place the PDMS stamp underneath the port-containing plexi-glass part and align with PDMS port features.
 - a. Hold and invert the top plexi-glass piece
 - b. Contact the PDMS with top plexi-glass
 - c. Hold PDMS and plexi-glass tight and invert back so that the PDMS is serves as a support
8. Place Dowel Pins in the ports of the top plexi-glass piece
 - a. They should just barely seal the port openings (not tight)
9. Using a 1mL syringe, inject about 300-400 µL of the collagen/cells mixture until it flows out through the outlet port of the gel loading ports.
 - a. Make sure to apply a slight pressure to the plexi-glass to prevent tipping and collagen leakage while injecting
10. Dispense 170 µL of collagen/cell mixture on top of the covers-slip.
11. Place the PDMS slab on top of the collagen/cell mixture
 - a. Make sure that all of the collagen/cell mixture has spread evenly throughout the surface of the cover-slip
12. Place both collagen/cell mixture-containing pieces inside the incubator for 25 mins.
13. Hold the PDMS and the plexi-glass with both hands to rotate again. Drop the Dowell pins and place the plexi-glass on the dish that contained it.

14. Dispense GM (pre-incubated at 37°C and 5%CO₂) around the edges of the PDMS slab on the cover-slip and the PDMS stamp with the top plexi-glass piece.
15. Gently pull off the PDMS stamp along the featured channels length and dispense GM on the collagen to prevent drying and promote cell viability (pH = 7.4).
16. Gently lift up a corner of the PDMS slab to allow GM to enter in the sealed gap. Very gently start push sideways to remove the PDMS lab. It should float once it completely comes off.
17. Invert the bottom plexi-glass with collagen layer on the cover-slip.
18. Place two screws on the two holes nearest to you.
19. Align the screws with their appropriate holes in the inverted top plexi-glass piece.
20. Gently place down the inverted bottom plexi-glass piece along the featured channels length. The GM should exhibit a displacing front behavior until the two pieces meet completely.
21. Place the two remaining screws in the empty holes.
22. Twist the screws until they make minimal resistance. This is crucial to avoid channel squashing.
23. Invert the assembled device and inspect for channel fidelity.
24. Feed the reservoirs with GM and let stabilize.
25. Prepare cells that will make an endothelium to the desire working density.
26. Feed 10-20uL of the cell suspension using a pipette.
27. Let the cells attach to the collagen walls for an hour (minimum).
28. Replace GM with Vasculogenesis Media (VM).
29. Replenish VM every 2 days.

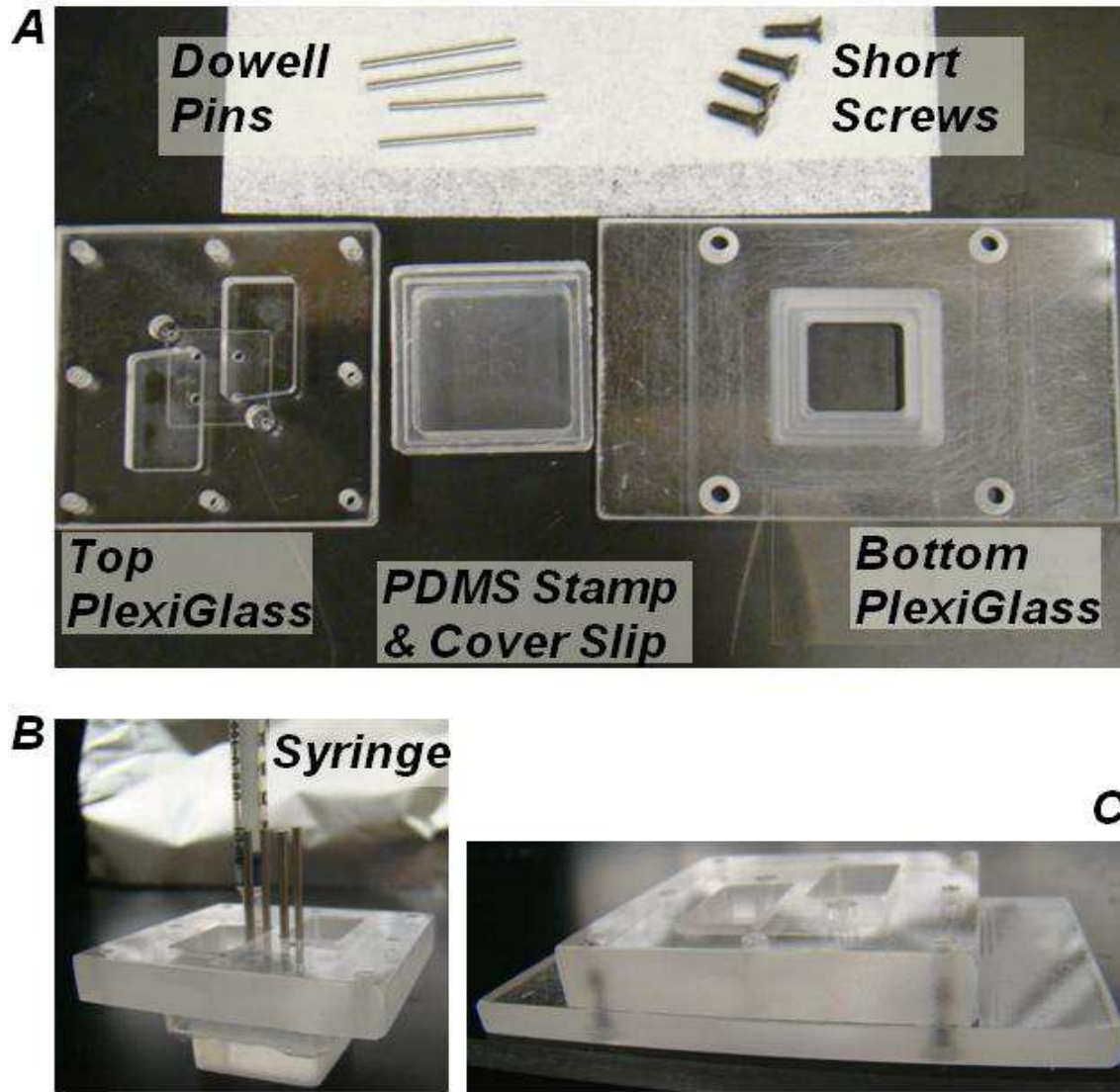


Figure A4. Perfuseable In Vitro Living Tissue (PiVLT) assembly depiction. (A) The different components used to setup the PiVLT. A 25 mm x 25 mm cover slip is reclined onto the PDMS stamp. The Bottom PlexiGlass part has a thin trench to accommodate the cover slip. (B) To create microfluidic features with collagen, the collagen is injection-molded into a setup that constitutes the PDMS stamp pressed onto the Top PlexiGlass and the Dowell Pins inserted into the appropriate ports. The syringe has a specialized port. (C) The final assembly has the Bottom PlexiGlass with the collagen-coated coverslip inserted in the trench, screwed onto the Top PlexiGlass.

APPENDIX B

MODELING THE CRYSTALLIZATION OF HYDROXYAPATITE IN A DOUBLE DIFFUSION SYSTEM

B-1 INTRODUCTION

Calcium phosphate (CP) is a major component of bone and teeth [1-2] and exists as a biogenic crystalline phase, hydroxyapatite (HA; $\text{Ca}_{10}(\text{PO}_4)_6\text{OH}_2$) [2-3] *in vivo*. The formation of HA, via a process of biomineralization, is biologically regulated, leading to the creation of organic-inorganic composites (e.g., bone-cartilage interface; Fig.1); a transition from compliant cartilage to biomineralized bone is made through an intermediate continuous region called calcified cartilage. This transition region is known as the tidemark between bone and cartilage. The osteochondral interface typically occurs over a sub-millimeter scale and has mechanical properties that bridge those of the two dissimilar tissues. It has been proposed that *in vivo* concentration gradients of inhibitors and promoters are involved in regulating the formation of the bone mineral, HA [4-7]. Such gradients of biomacromolecules may establish the location of the tidemark and contribute to the successful integration of cartilage with bone. For example, articular chondrocytes are known to synthesize pyrophosphate (PPi), which is a soluble and potent inhibitor of the HA formation [7-9]. If the balance of interactions between the biomacromolecules and tissue matrices is broken under certain pathological conditions, abnormal biomineralization can occur and cause detrimental effects in tissues and organs [6-7]. Therefore, it is important to study the biomineralization to identify matrix molecules involved and elucidate their roles. Improved understandings would also lead to success in designing new organic-inorganic composite materials that could be used for tissue engineering and therapeutic purposes.

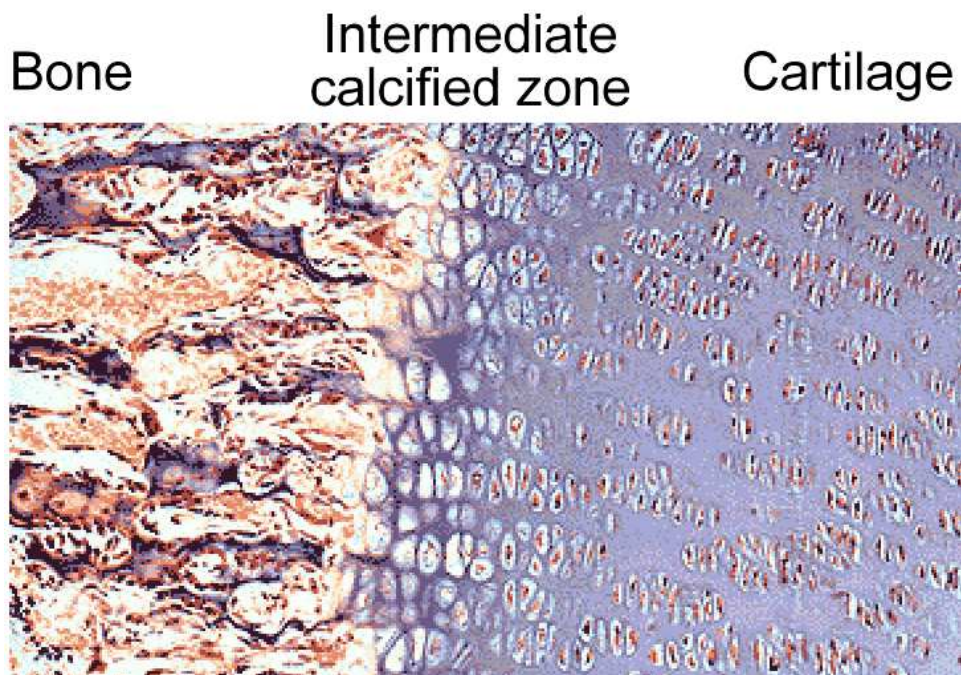


Figure 1. Histological section of bone-cartilage interface. Image was obtained from <http://faculty.une.edu/com/abell/histo/histolab3.htm>.

In understanding the mineralization process of HA, *in vitro* hydrogel precipitation methods, used for the past decades for the growth of large crystals [10], have been applied to elucidate the roles of biomacromolecules [11]. Such *in vitro* hydrogel precipitation methods in the absence of cells aid to characterize the chemical process in a more controlled manner, compared with *in vivo* studies. In addition, the hydrogel precipitation methods have advantages that solution precipitation methods do not provide: 1) small amounts of biomolecules are required to reduce the volume of precipitation, and 2) diffusion-reaction of molecules and ions in gels allows the recapitulation of physiological environment. Silverman and Boskey introduced the double-diffusion system (DDS), where calcium ions are delivered from one end and phosphate ions from the other, for the purposes described above [11] (also see Fig.2).

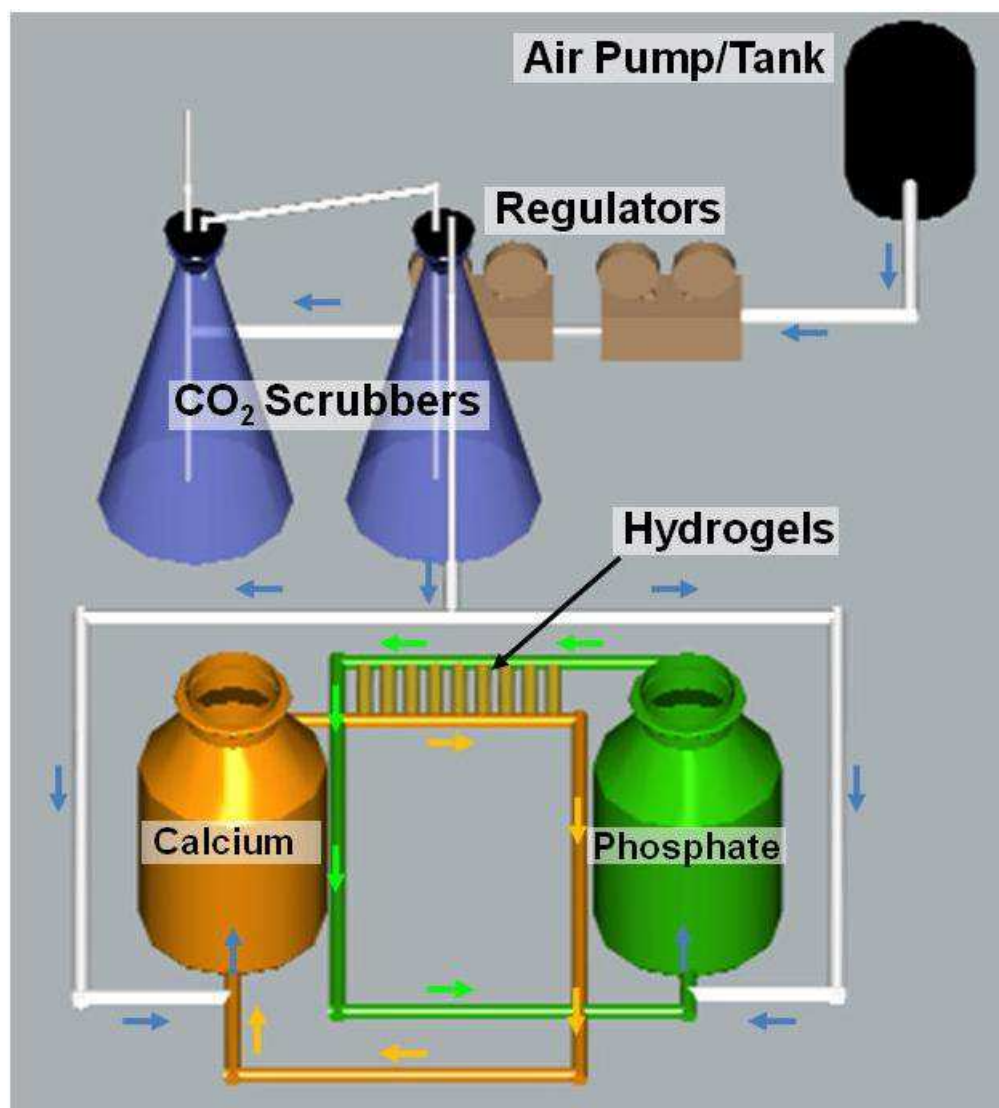
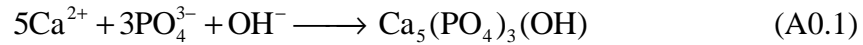


Figure 2. Schematic diagram of the dynamic hydroxyapatite formation system (the double-diffusion system (DDS)). Image was obtained from Dorvee, J.R., Boskey, A.L., and Estroff, L.A. Rediscovering Hydrogel-Based Double-Diffusion Systems for Studying Biomineralization. *Calcif. Tissue Int.* **75** (6), 494-501 (2004).

Along with experimental observations, it would be also valuable to model the formation of HA (i.e., precipitation of calcium phosphate) in the DDS because models can be applied in a predictive role. Using the models as our guide, we could predict: 1) when a precipitation event will occur, 2) where it will be within a hydrogel, and 3) the expected composition of the mineralized band. However, the formation of HA is believed to be mediated by the formation of intermediate CP precursors such as amorphous calcium phosphate (ACP), dicalcium phosphate dihydrate(DCPD), and octacalcium phosphate (OCP)[2-3, 12]. In addition, it is a complex multi-step process with seemingly high order kinetics (i.e., 9th order with respect to calcium, phosphate, and hydroxyl ions) where supersaturation, nucleation, and crystal growth are involved [12-14]. The global reaction for the crystallization of HA in aqueous environment is given by



Despite a large number of both experimental and theoretical studies aimed to decipher details about the mechanisms and kinetics of the formation of HA[12-17], most studies are based on the solution precipitation methods. From the perspective of mass transport, these studies have a fundamental difference; in the solution precipitation methods, reacting ions are convectively well-stirred whereas, the DDS is a continuously fed unstirred reactor (CFUR) that results in the creation of spatially distinct regions (i.e., regions for the free diffusion of the ions and a region for the fast crystal formation). This complicated nature of the formation of HA and the lack of necessary information (e.g., interfacial tension between HA and a gel matrix) make it difficult to model the system. Furthermore, although a supersaturation threshold can be identified experimentally (i.e., critical concentrations of calcium and phosphate ions at a given position and

time) and has been widely used to predict the formation of HA for decades[18], it has not led to the identification of a specific reaction mechanism in the DDS.

In order to generate predictions of the dynamics and mass transport of the system, our attempt to use continuum equations to simulate the concentration profiles confronted divergence limitations due to the complex nature of the high order kinetics of the formation of HA embedded in supersaturation (S); the continuum model is extremely unstable due to the high local variability of supersaturation, which is the driving force for crystal growth. This model is based on the assumption that the rate-limiting step follows the global stoichiometry that is high-ordered (Eq.1). We note that orders higher than three for an elementary reaction are extremely rare in homogeneous chemical reactions because the kinetic order of reactions is directly dependent on the probability of physical contact between the reacting species; the probability for quadrimolecular collisions to occur in a single step is negligibly small. Such high orders may become probable in heterogeneous reactions such as adsorption or crystal growth where multimolecular aggregates and multistep processes are involved. A different approach that can be considered is to simulate all the dynamics (chemical and physical) by the implementation of Kinetic Monte Carlo (KMC) method. Along with Molecular Dynamics (MD), this approach has been widely used to understand crystal growth phenomena on a molecular level, by treating the growth process as a stochastic and probabilistic description [19-20]. Application of Kinetic Montecarlo (KMC) Methods to large scale systems is mainly limited by its computational cost, making the transition from the molecular scale to the mesoscopic scale technically challenging. Modified KMC methods, like coarse-grained KMC and spatially adaptive coarse-grained [21-22], have been developed for improved computational cost, permitting stochastic simulations even at mesoscopic level.

In this work, we report a computational model that presents an alternative solution to the limitations. A multiscale computational modeling approach, by hybridizing KMC with finite difference (FD), offers the advantage of separating the diffusion time scale (treating it as a bulk property of the system) from the reaction time scales of nucleation and growth. In addition, the discretization of the reaction tube in the DDS (also see Fig. 5A) enables the evaluation of local dynamics. It is important to note that similar multiscale approaches have been previously applied to diffusion-limited systems [21-22], although to the best of our knowledge this is the first time this approach is applied to a crystallization problem. This technique allows modeling the formation of HA under the assumption of stoichiometric mechanisms of the crystallization. We focus on finding the location of a confined reaction layer (i.e., the depletion sink) and its characteristics. We also describe the use of our model to include inhibition of HA formation. We then conclude with our current assessment of the computational models as predictive tools and future directions towards exploiting the biomineralization processes.

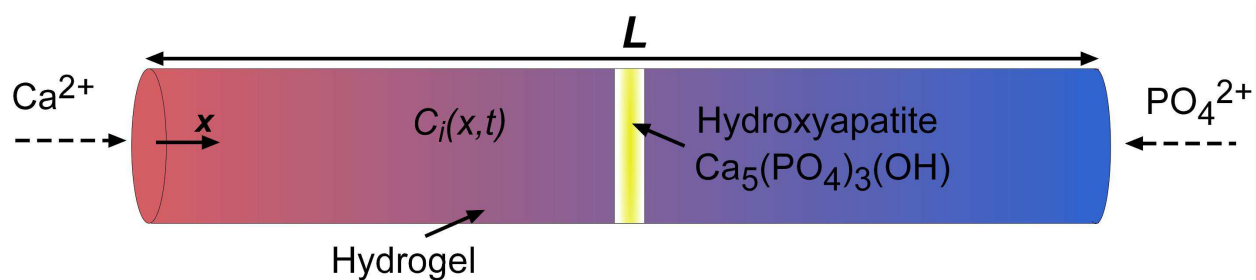
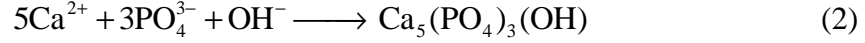


Figure 3. Schematic representation of double-diffusion system. The formation of hydroxyapatite (HA; yellow). The blue shading represents the concentration of phosphate ions. The red shading represents the concentration of calcium ions. The white bands represent the depletion gradient of HA towards the feeds of the DDS.

B-1 Formation of HA

The global reaction for the crystallization of HA in aqueous environment is given by



However, the crystallization process is known to occur by two distinct steps: nucleation and growth (Fig. 4).

A. Nucleation

The model for nucleation rate considered is based on the classical heterogeneous nucleation:

$$R_n = (K' P) \exp \left[-\frac{\beta \gamma_i^3 v^2 f(\theta)}{k_B^3 T^3 \ln(S)^2} \right] \quad (3)$$

β is a geometrical factor and in our model it has a value of $16\pi/3$ representing a spherical nucleus. γ_i is the interfacial tension between the calcium phosphate and the solution and has a value of 104 mJ/m^2 . v is the molecular volume of a HA unit determined by its crystal structure and has a value of 263.24 \AA^3 . $f(\theta)$ is the contact angle function given by the classical theory of heterogeneous nucleation[23], and it has values from 0 to 1. The contact angle function can be estimated based on the shape of HA on a substrate surface (e.g., gelatin or agarose). An angle of 0° corresponds to a value of the contact angle function of 0, while an angle of 180° corresponds to a value of 1 and represents homogeneous nucleation. We have chosen the contact angle function to have a value of $1/2$ since its variation does not affect significantly the nucleation

rate. k_B is Boltzmann's constant and T is the solution temperature (taken as ambient). K' is a constant that was determined from experiments and has a value of $1.364 \times 10^{28} L^{-1} s^{-1}$. The parameters described above are based on a theoretical study of calcium phosphate precipitation in simulated body fluid, a solution with calcium and phosphate ions, and a physiological pH value (i.e. pH 7.4)[13]. Supersaturation of HA, S_{HA} is defined as

$$S_{HA} = \frac{a_{Ca^{2+}}^5 a_{PO_4^{3-}}^3 a_{OH^-}}{K_{sp,HA}} \quad (4)$$

where a represents activity of each ion and K_{sp} (2.35×10^{-59}) is the solubility product for HA crystal. The activities are defined as the product of the molar fraction and the activity coefficient for each ion. The activity coefficients (γ) for each ion can be obtained from the modified Debye-Hückel equation:

$$\log \gamma_i = -A z_i^2 \left(\frac{I^{1/2}}{1 + I^{1/2}} - 0.3I \right) \quad (5)$$

where A is the Debye-Hückel constant dependent on temperature ($A = 0.5211$ at 37°C), z_i is the charge number of the ion and I is the total ionic strength of the solution and is defined as

$$I = \frac{1}{2} \sum c_i z_i^2 \quad (6)$$

where c_i is the molar concentration of each ion unit. OH^- concentration is always fixed by unchanging pH. The nucleation rate is also proportional to the probability that the appropriate ion units of calcium phosphate meet to compose a nucleus in solution. This probability is defined as

$$P = \frac{9! [Ca^{2+}]^5 [PO_4^{3-}]^3 [OH^-]}{5!3! ([Ca^{2+}] + [PO_4^{3-}] + [OH^-])^9} \quad (7)$$

B. Crystal growth

The model for crystal growth rate is proportional to the relative solution supersaturation, σ , with respect to HA, defined as

$$R_{cg} = k_{cg} \sigma^n \quad (8)$$

where n is the apparent order of the crystallization process and k_{cg} is the rate constant. Both if these values are determined by fitting experimental data of crystal growth ($n = 1.87$ and $k_{cg} = 2.59201 \times 10^{22} \text{ m}^{-2} \text{ s}^{-1}$) [14]. The relative supersaturation is defined as

$$\sigma = S^{1/9} - 1 \quad (9)$$

where S is the supersaturation as in Eq.3.

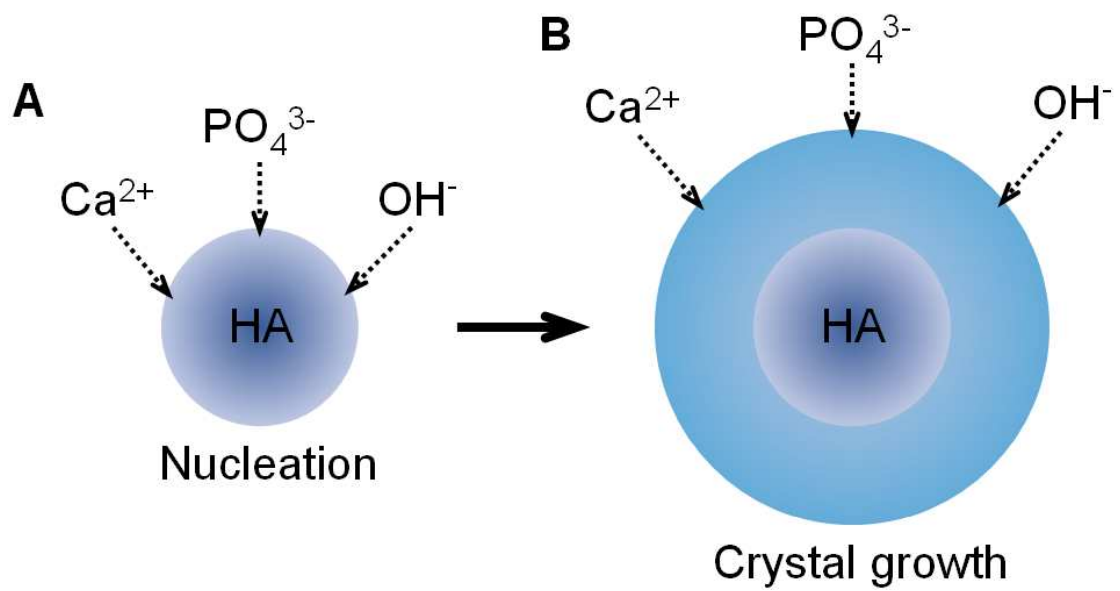


Figure 4. Schematic illustration of the formation of HA. A. Nucleation. B. Crystal growth.

B-2 Implementation of multiscale computational modeling: finite difference (FD)-Kinetic Monte Carlo (KMC)

Multiscale computational modeling approaches can be used for 1) metal electrodepositing on a surface[24] or 2) a fluid phase in bulk and catalytic surface reactions within catalytic flow reactors[25], where macroscopic modeling at a continuum level does not fully capture near-surface chemistry and detailed reaction mechanisms. Such multiscale modeling approaches are carried out by coupling continuum-level modeling with molecular level simulations.

In order to describe both the diffusion of calcium and phosphate ions in the bulk of the DDS, and the crystallization process of HA, we have developed similar multiscale modeling by coupling finite difference (FD) and Kinetic Monte Carlo (KMC). Then, the diffusion of the ions is modeled on a macro-scale while the nucleation and crystal growth are modeled on a micro-scale. The diffusion equation

$$\frac{\partial c_i}{\partial t} = D_i \frac{\partial^2 c_i}{\partial x^2} \quad (10)$$

has been solved for each ion in the system using the Crank-Nicholson implicit method, dividing the double-diffusion system in several volume elements of length Δx (Fig.5). The transient 1-D diffusion equation becomes

$$D_i \left\{ \frac{c_{m+1,i} - 2c_{m,i} + c_{m-1,i}}{(\Delta x)^2} \right\} + R_{m,i} = \frac{\Delta c_{m,i}}{\Delta t} \quad (11)$$

where m is the element and i is the ion species. Since there is no reaction (due to decoupling), the R term is zero for all positions except for the first and last elements where R corresponds to the

boundary conditions (constant boundary concentrations). Since equation 11 is dependent on the concentrations at the m element, $m-1$ element, and $m+1$ element, we obtain a system of linear equations that can be solved using matrices:

$$\begin{bmatrix} \frac{-2D_i}{(\Delta x)^2} & \frac{D_i}{(\Delta x)^2} & & \\ \frac{D_i}{(\Delta x)^2} & \frac{-2D_i}{(\Delta x)^2} & \frac{D_i}{(\Delta x)^2} & \\ & \vdots & \vdots & \\ & \dots & \frac{D_i}{(\Delta x)^2} & \frac{-2D_i}{(\Delta x)^2} \end{bmatrix} \times \begin{bmatrix} C_{1,i} \\ C_{2,i} \\ \vdots \\ C_{m,i} \end{bmatrix} + \begin{bmatrix} \frac{D_i}{(\Delta x)^2} C_{0,i} \\ 0 \\ \vdots \\ \frac{D_i}{(\Delta x)^2} C_{m+1,i} \end{bmatrix} = \begin{bmatrix} \frac{\Delta C_{1,i}}{\Delta t} \\ \frac{\Delta C_{2,i}}{\Delta t} \\ \vdots \\ \frac{\Delta C_{m,i}}{\Delta t} \end{bmatrix} \quad (12)$$

The result is a matrix of the variation of concentration with respect to time for each single element, so we can predict the concentration profile of the microchannel after a time Δt .

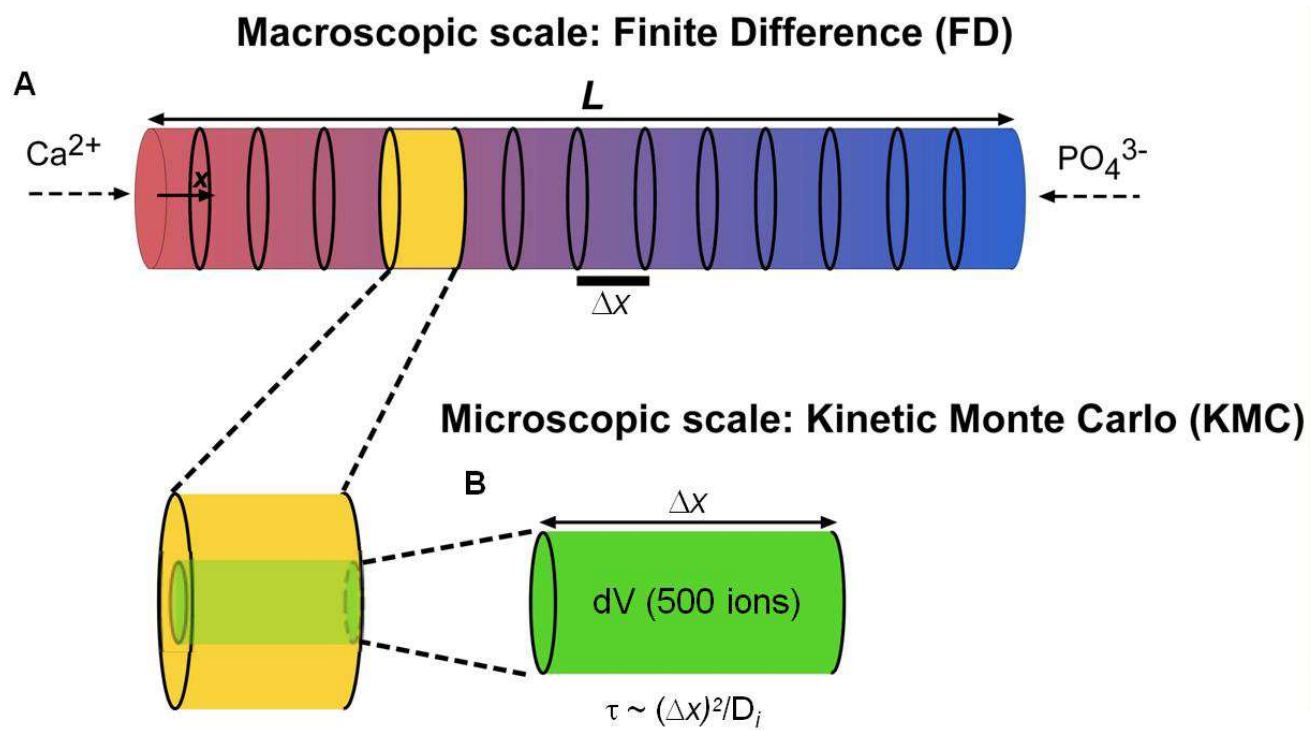
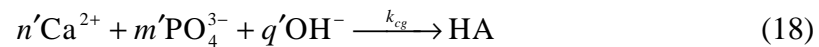
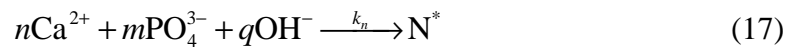
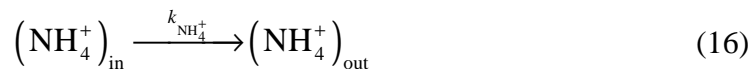
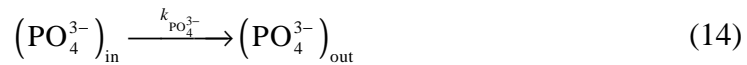
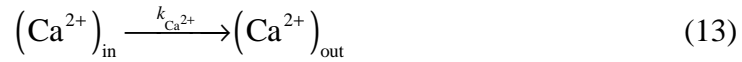


Figure 5. Schematic representation of the multiscale modeling, FD-KMC, showing macro to micro-scale transition. A. FD in macroscale. B. KMC in microscale.

Figure 5 depicts how a single volume element is considered as a perfect Stirred Tank Reactor (STR), yet this element is still too large to perform KMC. Therefore, it is divided into many sub-volumes of the same length (Δx); their radius is small enough that a concentration of 0.1 mol/L (reference concentration) corresponds to only 500 ions. This additional step is not preferable but necessary to lower computational load. Ideally, KMC could capture stochastic concentration fluctuations of ions within the entire single volume element, leading to increase of sampling species. In addition, radial symmetry is assumed within each element; the radial flux of any ion is the same towards and out of a sub-volume, such that only axial diffusion should be considered. The characteristic time of such diffusion (for any single ion) has been calculated from Einstein's result for Brownian motion. Then, we can consider 6 pseudo-reactions for the KMC approach:



If we consider, for example, the double-diffusion system divided in 15 elements, there are a total of 60 pseudo-reactions competing on the KMC and one is chosen randomly, based on their

propensity functions (Gillespie's method). A reaction time is sampled using KMC criteria.

Because HA cannot precipitate until Ca^{+2} and PO_4^{-3} encounter each other, the algorithm only takes diffusion into account until Ca^{+2} and PO_4^{-3} concentrations, at any single point, reach a value that makes nucleation feasible. Once they encounter, the algorithm makes a scale change in order to evaluate the probability of nucleation or growth following the KMC criteria.

Even though KMC is useful to deal with nucleation and growth pseudo-reactions, it is not practical to model diffusion using KMC, because of the size of the double-diffusion system. Therefore, we have linked KMC with FD when a diffusion pseudo-reaction is chosen based on randomly fluctuating probabilities; in other words, 1) no nucleation or growth occurs during the KMC sampled time that also randomly fluctuates, and 2) all the ions diffuse during the time interval based on the macroscopic diffusion with the Crank-Nicholson implicit method. On the other hand, if a nucleation or growth pseudo-reaction is chosen as a single event, no ion diffuses while the pseudo-reaction occurs.

The complexity of the model can be increased by adding a nucleation inhibitor to the system, which would occur if cells (chondrocytes) are seeded at a specific position of the double-diffusion system and start producing inhibitors such as pyrophosphate. As an example, our algorithm incorporates an inhibition factor (I^*) as a calcium ion sink:



B-3 Results and discussion

With input parameters summarized in Table 1, we have simulated the formation of HA, using the FD-KMC. Calcium and phosphate ions were allowed to diffuse from the two ends of

the DDS and to form HA. The location of the first crystal that is formed determines a “sink” for both, calcium and phosphate ions. However, we note that this does not imply that crystallization does not take place in neighboring locations (Fig. 6E). The computational results show reasonable agreement with experimental observations (Fig. 6A, B, and C). Figure 6E shows that HA crystals form slightly away from the center towards the phosphate side. Note that the simulation was run with experimentally measured diffusivity of calcium (i.e., $6 \times 10^{-10} \text{ m}^2/\text{s}$) and phosphate (i.e., $3.9 \times 10^{-10} \text{ m}^2/\text{s}$) ions within gelatin. Yet, HA forms at about the center of gelatin (Fig. 6A). Interestingly, in both experiment and simulation, the formation of HA is slightly shifted towards the phosphate side in agarose (Fig. 6B). These experimental observations suggest that crystal-matrix interactions may be involved in the formation of HA. In addition, the disagreement between the computational result and the experimental observation in gelatin is likely due to a possibility that input parameters such as interfacial surface tension, rates of nucleation and crystal growth could be different in gel matrices. The quoted values are reported ones that have been measured or estimated in solution.

In an effort to predict the effect of an inhibitor, we have placed a steady source of a calcium chelator at element 6 and allowed it to diffuse along the DDS. As expected, the presence of an inhibitor displaces the “sink” location, for both calcium and phosphate ions (Fig. 6D). The location of the inhibitor determines where the first and largest crystal forms. This can be explained by the fact that nucleation rates are orders of magnitude smaller than growth rates. The outcome of this situation is that once a nucleus is formed, crystal growth takes place very rapidly and drives the consumption of reacting ions. Figure 6E indicates that the number of HA crystals formed in the presence of an inhibitor is smaller than that without the inhibitor. This is because the crystallization reaction requires more phosphate ions than calcium for it to begin while the

inhibitor continuously chelates calcium ions. We have observed that locating the inhibitor (i.e. calcium chelator) at the phosphate side hardly affected the formation of HA (data not shown). These computational results suggest that once specific roles of inhibitors are identified, we could control the formation of HA in a spatially resolved manner.

Nucleation rates for the HA formation are highly dependent on supersaturation (see Eq. 3). This dependency confers the continuum approach unstable given the variable nature of the S along the DDS (Fig. 7). Because we are able to combine continuum-based diffusion calculations that provide data to a localized KMC, we can overcome these limitations.

Table 1. Input parameters used in the FD-KMC

Calcium boundary concentration	$c_{0, \text{Ca}^{2+}}$	0.1 mol/L
Phosphate boundary condition	$c_{0, \text{PO}_4^{3-}}$	0.1 mol/L
Hydroxyl ion concentration	c_{OH^-}	2.5×10^{-7} mol/L
Diffusivity of calcium ion	$D_{\text{Ca}^{2+}}$	6×10^{-10} m ² /s
Diffusivity of phosphate ion	$D_{\text{PO}_4^{3-}}$	3.9×10^{-10} m ² /s
Diffusivity of chloride ion[26]	D_{Cl^-}	2×10^{-9} m ² /s
Diffusivity of ammonium ion[26]	$D_{\text{NH}_4^+}$	1.9×10^{-9} m ² /s
Diffusivity of inhibitor	D_I	2×10^{-10} m ² /s
Rate of inhibitor production	R_I	5×10^{-7} mol/L · s
Rate constant for nucleation[13]	K'	1.36×10^{-28} 1/L · s
Interfacial surface tension between HA and water[13]	γ_I	0.104 J/m ²
Molecular volume of HA[13]	v	2.63×10^{-28} m ³ /mol
Contact angle function[13]	f	0.5
Solubility product of HA[13]	$K_{sp, \text{HA}}$	2.35×10^{-59}
Rate constant for crystal growth[14]	k_{cg}	2.59×10^{22} 1/m ² · s

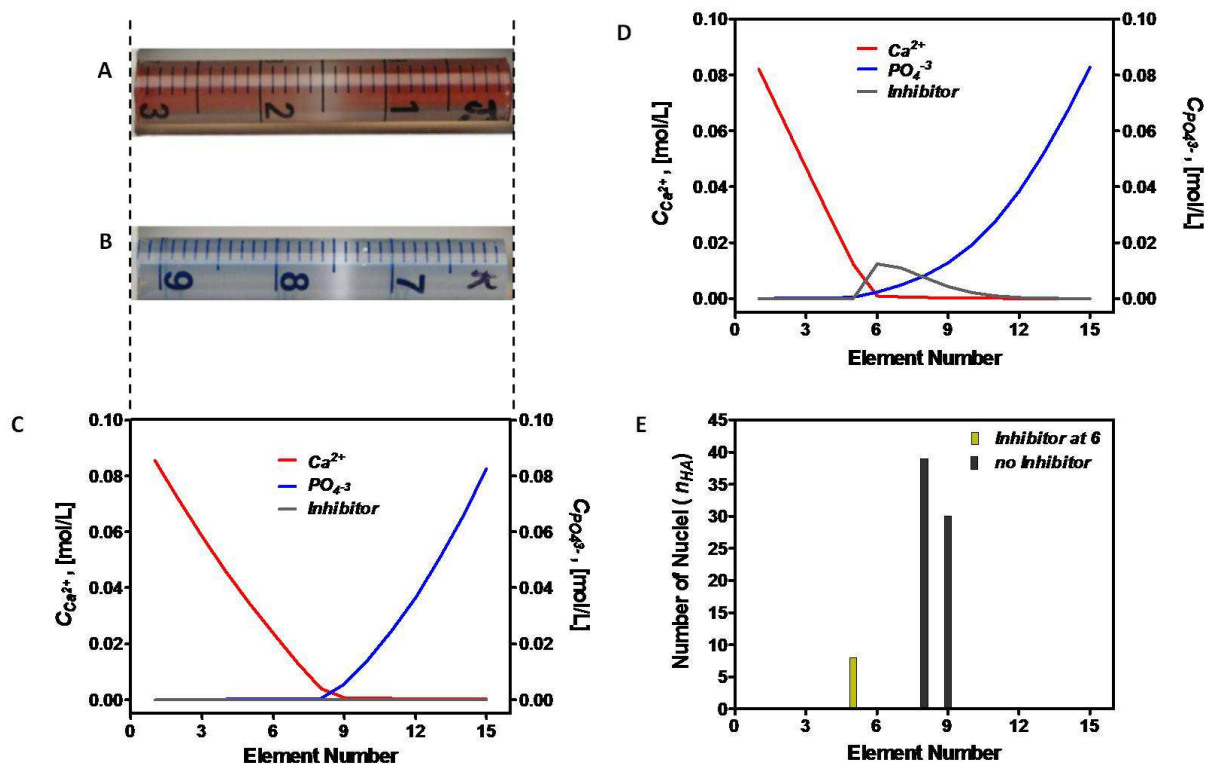


Figure 6. Formation of HA in the DDS: experimental observations and FD-KMC results. A and B. Optical micrographs showing the formation of HA within gelatin (A) and agarose (B) on day 5. **C.** Spatial distribution of ions in the absence of an inhibitor on day 5. **D.** Spatial distribution of ions in the presence of an inhibitor (a Ca^{2+} chelator produced at element 6) on day 5. **E.** Spatial distribution of HA nuclei with and without the inhibitor.

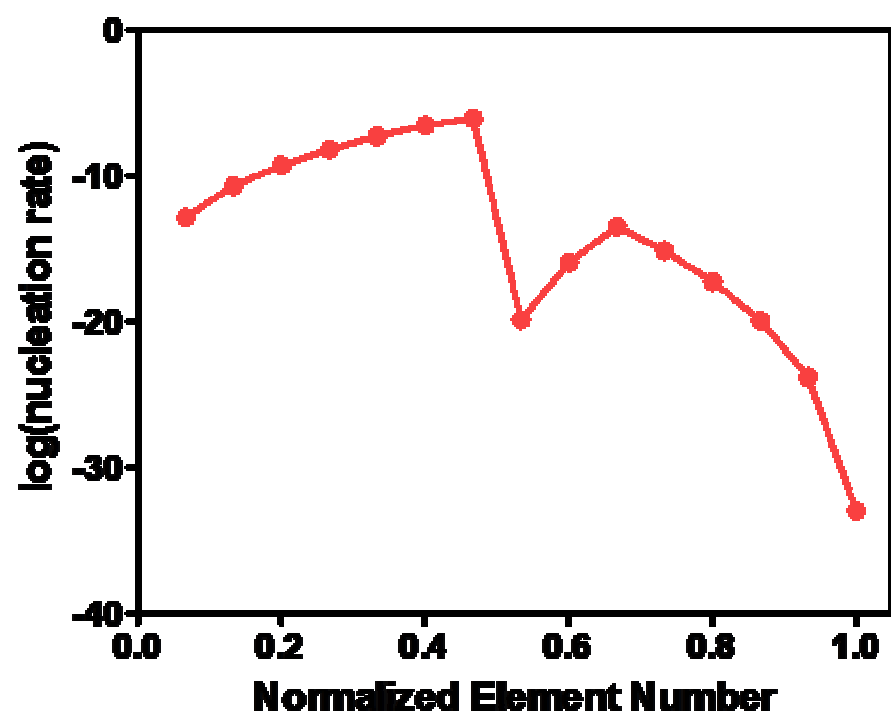


Figure 7. Nucleation rates vary in orders of magnitude for the formation of HA.

B-4 Conclusions

We have successfully initiated a computational model for the formation of hydroxyapatite in the double-diffusion system. Our multiscale FD-KMC model demonstrated the potential to model the high order crystallization process. This technique could serve as a useful method to validate and predict experimental observations. In addition, our algorithm is flexible enough to make modifications for the addition of an inhibitor or a promoter, and pre-existing nuclei sites. Yet, the FD-KMC is seemingly less powerful largely due to its application to the formation of HA in the DDS that has not been fully understood; for example, input parameters have not measured specifically in our gel matrices. We note that such measurement still remains a challenge. Exploration of parameter space with this FD-KMC could aid the investigation of kinetic information of biomineralization processes.

B-5 References

1. Estroff, L.A., *Introduction: Biomineralization*. Chemical Reviews, 2008. **108**(11): p. 4329-4331.
2. Wang, L.J. and G.H. Nancollas, *Pathways to biomineralization and biodemineralization of calcium phosphates: the thermodynamic and kinetic controls*. Dalton Transactions, 2009(15): p. 2665-2672.
3. Mahamid, J., et al., *Amorphous calcium phosphate is a major component of the forming fin bones of zebrafish: Indications for an amorphous precursor phase*. Proceedings of the National Academy of Sciences of the United States of America, 2008. **105**(35): p. 12748-12753.
4. Boskey, A.L., *Matrix proteins and mineralization: An overview*. Connective tissue research, 1996. **35**: p. 357-363.
5. Furedimilhofer, H., et al., *Interactions of matrix proteins from mineralized tissues with octacalcium phosphate*. Connective Tissue Research, 1994. **30**(4): p. 251-264.
6. Steitz, S.A., et al., *Osteopontin inhibits mineral deposition and promotes regression of ectopic calcification*. American Journal of Pathology, 2002. **161**(6): p. 2035-2046.
7. Thouverey, C., et al., *Inorganic pyrophosphate as a regulator of hydroxyapatite or calcium pyrophosphate dihydrate mineral deposition by matrix vesicles*. Osteoarthritis and Cartilage, 2009. **17**(1): p. 64-72.
8. Kalya, S. and A.K. Rosenthal, *Extracellular matrix changes regulate calcium crystal formation in articular cartilage*. Current Opinion in Rheumatology, 2005. **17**(3): p. 325-329.

9. Terkeltaub, R.A., *Inorganic pyrophosphate generation and disposition in pathophysiology*. American Journal of Physiology-Cell Physiology, 2001. **281**(1): p. C1-C11.
10. Henisch, H.K., *Crystals in gels and liesegang rings: in vitro veritas*. 1988, Cambridge: Cambridge University Press.
11. Silverman, L. and A.L. Boskey, *Diffusion systems for evaluation of biomineralization*. Calcified Tissue International, 2004. **75**(6): p. 494-501.
12. Ball, V., et al., *Nucleation kinetics of calcium phosphates on polyelectrolyte multilayers displaying internal secondary structure*. Crystal Growth & Design, 2006. **6**(1): p. 327-334.
13. Lu, X. and Y. Leng, *Theoretical analysis of calcium phosphate precipitation in simulated body fluid*. Biomaterials, 2005. **26**(10): p. 1097-1108.
14. Dalas, E., et al., *The effect of Leucine on the crystal growth of calcium phosphate*. Journal of Materials Science-Materials in Medicine, 2008. **19**(1): p. 277-282.
15. Onuma, K., et al., *Precipitation kinetics of hydroxyapatite revealed by the continuous-angle laser light-scattering technique*. Journal of Physical Chemistry B, 2000. **104**(45): p. 10563-10568.
16. Cao, X., et al., *Inhibition of calcium phosphate precipitation under environmentally-relevant conditions*. Science of the Total Environment, 2007. **383**(1-3): p. 205-215.
17. Spanos, N., et al., *Precipitation of calcium phosphate from simulated milk ultrafiltrate solutions*. Crystal Growth & Design, 2007. **7**(1): p. 25-29.
18. Pucar, Z., B. Pokric, and A. Graovac, *Precipitation in gels under conditions of double diffusion - critical concentrations of precipitating components*. Analytical Chemistry, 1974. **46**(3): p. 403-409.

19. Kotrla, M., *Numerical simulations in the theory of crystal growth*. Computer physics communications, 1996. **97**: p. 82-100.
20. Levi, A.C. and M. Kotrla, *Theory and simulation of crystal growth*. Journal of Physics-Condensed Matter, 1997. **9**(2): p. 299-344.
21. Gill, S.P.A., P.E. Spencer, and A.C.F. Cocks, *A hybrid continuum/kinetic Monte Carlo model for surface diffusion*. Materials Science and Engineering a-Structural Materials Properties Microstructure and Processing, 2004. **365**(1-2): p. 66-72.
22. Zheng, Z.M., et al., *A hybrid multiscale kinetic Monte Carlo method for simulation of copper electrodeposition*. Journal of Computational Physics, 2008. **227**(10): p. 5184-5199.
23. Mullin, J.W., *Crystallization*. 4th ed. 2001, Oxford: Butterworth-Heinemann.
24. Pricer, T.J., M.J. Kushner, and R.C. Alkire, *Monte Carlo simulation of the electrodeposition of copper - I. Additive-free acidic sulfate solution*. Journal of the Electrochemical Society, 2002. **149**(8): p. C396-C405.
25. Majumder, D. and L.J. Broadbelt, *A multiscale scheme for modeling catalytic flow reactors*. Aiche Journal, 2006. **52**(12): p. 4214-4228.
26. *CRC Handbook of Chemistry and Physics*, D.R. Lide, Editor.

APPENDIX C

SUPPLEMENTARY FIGURES FOR CHAPTER 2 AND CHAPTER 3

Here I include figures that serve to supplement the data discussed in **Chapter 2** and in **Chapter 3**. The figure is associated to particular sections in the chapters; they are specified in the caption of each figure. If the supplemental figure required an experimental setup not described before it can be found here.

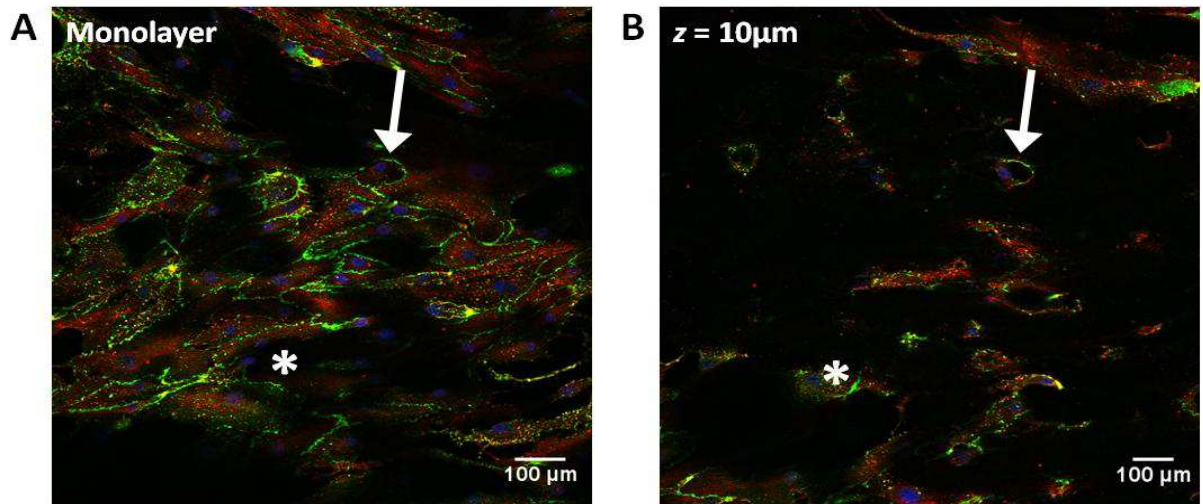


Figure C1. Monolayer deformations in invasion experiments. The context for this figure is found in section 2.3.1.3 of Chapter 2. **(A)** Confocal micrograph of HUVECs seeded on top of collagen. After 7 days of culture invasions were found (section 2.3.1.2, Fig. 5, Chapter 2; arrowheads). Cells remaining on the surface of the collagen exhibit cell-cell adhesion and are flat as seen by the extensive stain for membrane protein CD31 (green) and for Actin (red), respectively. **(B)** Same region found in (A), 10 μm below the monolayer. Blue is DAPI staining nuclei. Sections of the monolayer are found at $z = 10 \mu\text{m}$ from monolayer (white star) indicating a non-uniformity of the monolayer on the order of 10 μm.

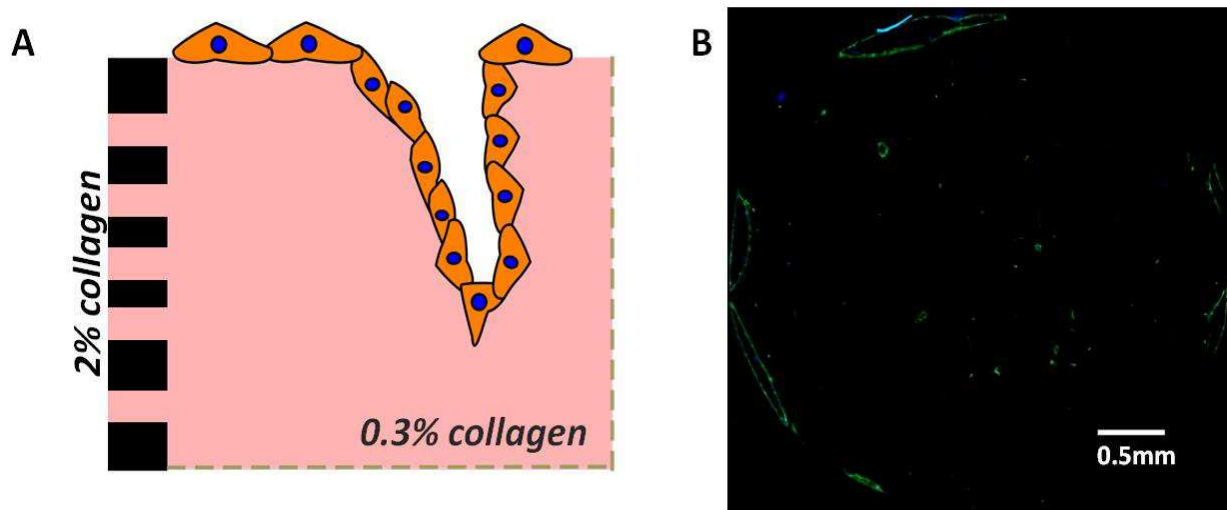


Figure C2. Angiogenic invasions near permeable boundary. The context for this figure is found in section 2.3.1.3 of Chapter 2. In order to discriminate chemical or mechanical effects on vascular morphogenesis, we considered a rigid permeable boundary which allows the passage of mass and thus does not create a chemical image for a cell or sprout in its vicinity. The solid permeable boundary weakens the reflection of solutes but not the reflection of stresses. We used this system to discriminate mechanics from mass transfer in cell communication: if cells or invasions near a permeable boundary would show a decrease in a biased behavior, then this behavior would have been dominated by mass transfer. **(A)** Collagen at a concentration of 2% serves as a permeable boundary. The concentration of collagen in the bulk of the well is 0.3%. An angiogenic invasion (multi-cellular event) is seen near the wall. **(B)** Confocal fluorescence micrographs of invasion assays using well-diameters of 4 mm at 50 μm below the monolayer. Green is actin, and blue is DAPI staining nuclei.

1. Preparation of collagen wells; Fig C1 experimental procedure

Figure 2 in Chapter 3 depicts the process steps involved in making PDMS wells for the vertical anastomosis assay. We used an analogous procedure to fabricate collagen wells as illustrated in **Fig. C3**. Collagen at a concentration of 2% was used as a permeable boundary. Collagen solution was poured onto a cold plexi-glass mold with an array of cylindrical posts 2-mm high and 4-mm in diameter. The mold had additional height to make a 2-mm thick base for each well. Collagen gelation took place at 37°C for 30 min. Wells were separated from the master mold. Individual wells were separated from the array using a biopsy punch with an 8-mm diameter opening. An individual well of 2% collagen was placed in the center of a well in a 6-well plate. Collagen wells positioned in 6-well plates were kept at 4°C before being loaded with 0.3% collagen. Collagen wells loaded with 0.3% collagen were immediately placed in the incubator for gelation at 37°C for 30 min. Cells were then seeded at a density of 250 cells/mm², allowed to settle for 3 hrs, then fed with cell media. Cell media was replenished every 2 days. Total experimental time was 7 days.

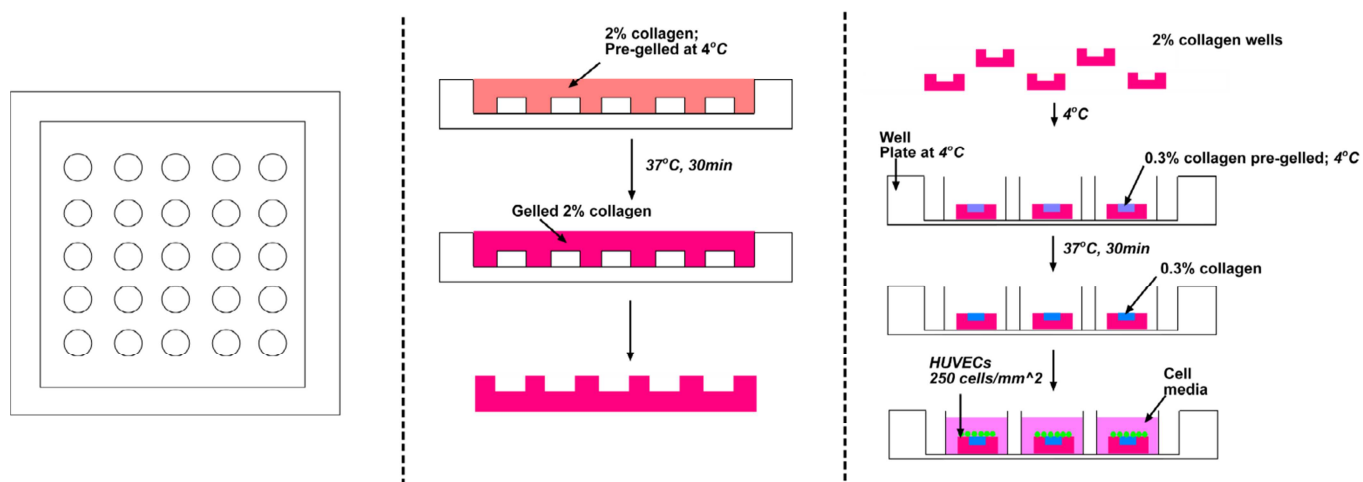


Figure C3. Preparation of collagen wells for invasion angiogenesis near permeable boundaries.

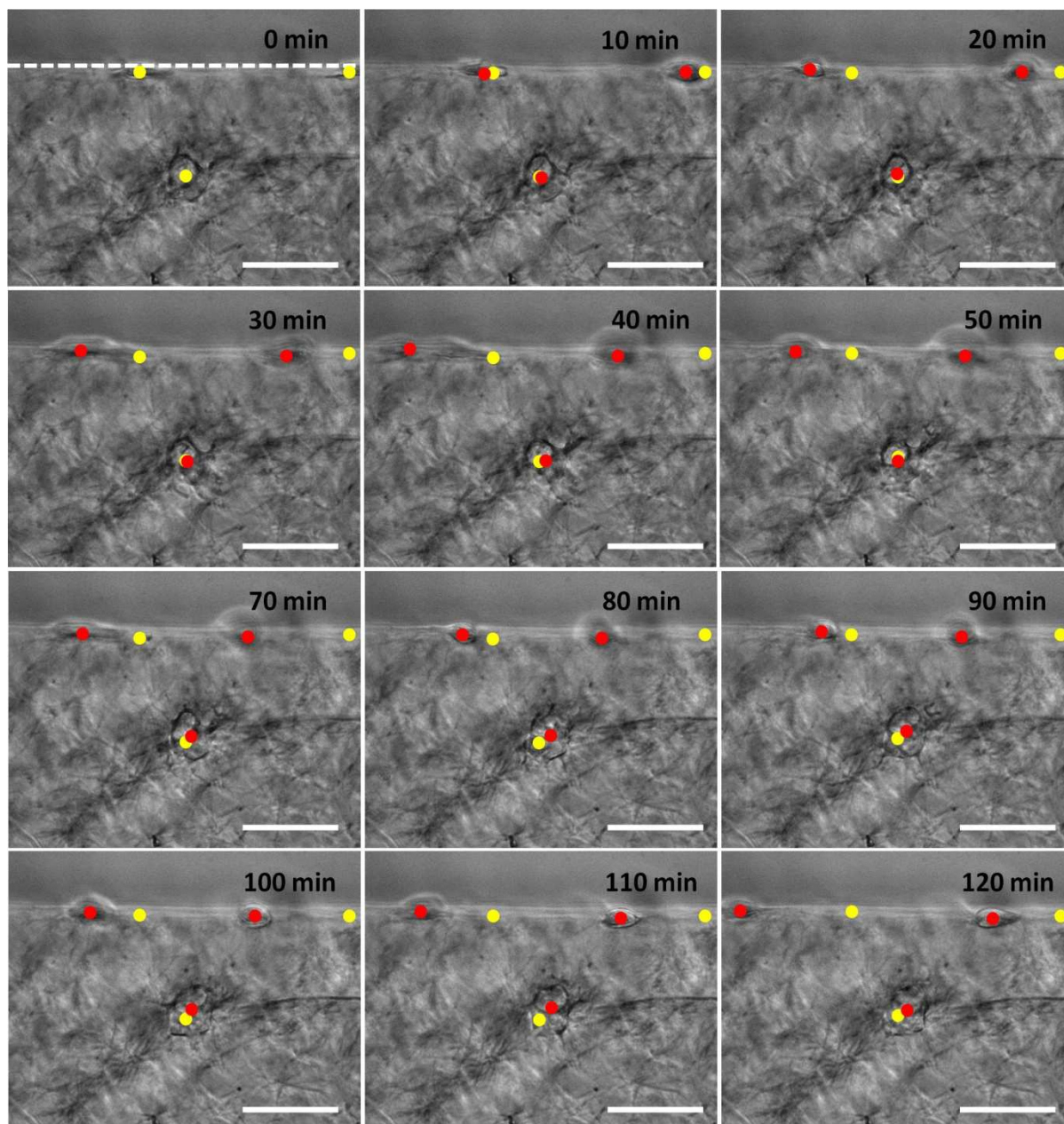


Figure C4. Qualitative migratory behavior of HUVECs in 2D versus 3D. The context for this figure is found in section 2.3.2.2 of Chapter 2. A brightfield micrograph was taken every 10 min at a magnification of 40x (see section 2.2.6 of Chapter 2 for methods). Two cells are found attached migrating on the boundary (dotted white line) and one cell is found fully embedded in the 3D collagen gel. The initial location of the center of each cell is represented by a yellow dot; the actual location at each time point in red. Scale bar: 50 μm .

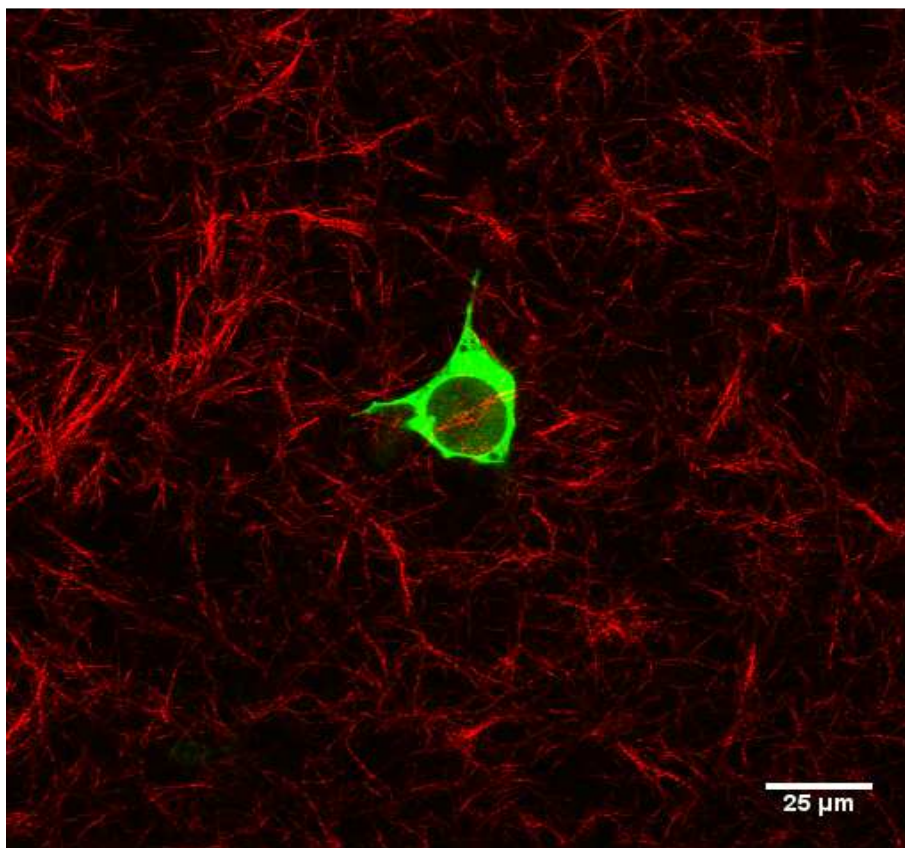


Figure C5. Concurrent visualization of cells and collagen fibers. The context for this figure is found in section 2. 4 of Chapter 2. An image of an individual HUVEC transduced to express GFP was superimposed with a reflectance microscopy image of collagen fibers (red; false color). The time difference between the acquisitions of both images was the time to switch lasers in the automated confocal microscope.

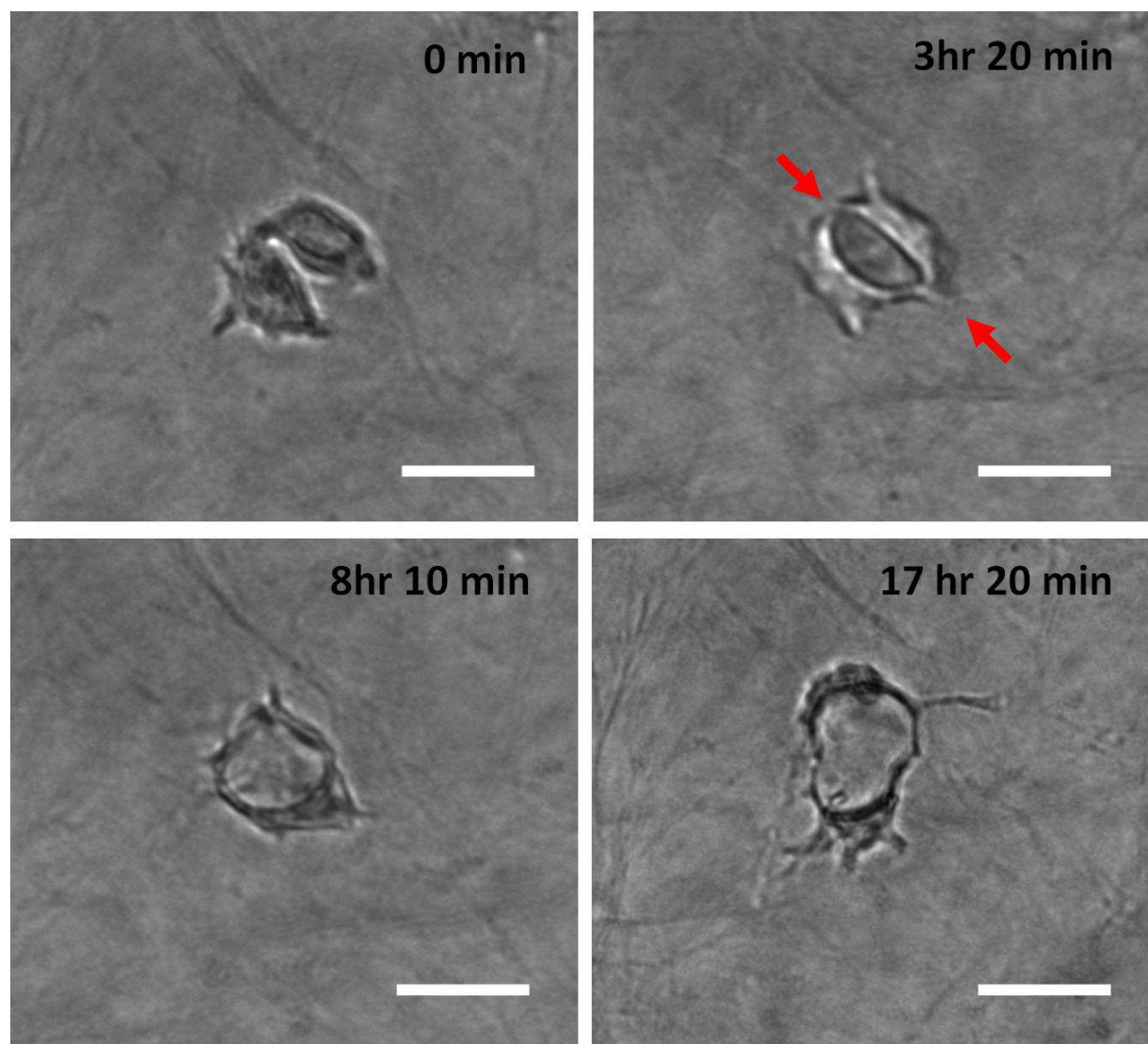


Figure C6. Connection of individual HUVECs during vasculogenesis. The context for this figure is found in section 2.4 of Chapter 2. A brightfield micrograph was taken every 10 min at a magnification of 40x (see section 2.2.6 of Chapter 2 for methods). Two cells are found in very close proximity to each other. Initial frame shows cells separated. Approximately 3.5 hrs later the two cells establish an anchor point among them (red arrows). Cells remain connected while maintaining their protrusive activity. Scale bar: 25μm.

APPENDIX D

MATLAB CODES

D-1 Spatial Analysis of Vertical Invasions

```
clear
clc

img_path0 = strcat('F:\Anthony EHD Research Files\Frit-Glass HUVEC Invasions\((ADS3_7-19-2012) Frit-Glass-
PDMS HUVEC INvasion Timepoints\Day7_4mm_Glass-6 (0, 10, 20, 50)');

img_path1 = strcat(img_path0, \*.tif');

file = dir(img_path1);
file_num = length(file);
file_name = {file.name};
file_date = {file.date};

cd(img_path0);

R1_count(file_num,1) = zeros; R2_count(file_num,1) = zeros; R3_count(file_num,1) = zeros; R4_count(file_num,1)
= zeros; %R5_count(file_num,1) = zeros; R6_count(file_num,1) = zeros;
F1_count(file_num,1) = zeros; F2_count(file_num,1) = zeros; F3_count(file_num,1) = zeros; F4_count(file_num,1)
= zeros; %F5_count(file_num,1) = zeros; F6_count(file_num,1) = zeros;
B1_count(file_num,1) = zeros; B2_count(file_num,1) = zeros; B3_count(file_num,1) = zeros; B4_count(file_num,1)
= zeros; %B5_count(file_num,1) = zeros; B6_count(file_num,1) = zeros;
nuclei_regions(4,file_num) = zeros;

RGB = imread(file_name{file_num});
%GS = rgb2gray(RGB);
%figure,imshow(GS),title('original GS')
R = RGB(:, :, 1);
%G = RGB(:, :, 2);
%B = RGB(:, :, 3);

% if f == 1

[IC RECT] = imcrop(R);
[rows,columns] = size(IC);
center_x = round(columns/2);
center_y = round(rows/2);

if rows < columns
    rmax = rows - center_x;
else
    rmax = columns - center_x;
end

pixel_length_conversion_4mm = 660;
pixel_lenght_conversion_1mm = 1173;
well_size = input('Is the well 4mm in diameter? (1 for yes/0 for no): ');
```

```

inner_radius = input('What is the inner radius (in mm) for the binning? ');
if inner_radius == 1
    ring_area = 3.1416; %mm^2
elseif inner_radius == 0.75
    ring_area = 1.7671; %mm^2
elseif inner_radius == 0.25
    ring_area = 0.1963; %mm^2
end

if well_size == 1
    rf = inner_radius*660;
else
    rf = inner_radius*1173;
end

r1 = rf; r2 = sqrt(2)*rf; r3 = sqrt(3)*rf; r4 = sqrt(4)*rf; %r5 = 5*rf; r6 = 6*rf;

for f = 1:file_num
    RGB = imread(file_name{f});
    R = RGB(:,:,1);
    B = RGB(:,:,3);
    R1 = 0; R2 = 0; R3 = 0; R4 = 0;

    IC = imcrop(R,RECT);
    ICa = im2bw(IC,0.2);

    for i = 1:rows
        for j = 1:columns
            r = sqrt((i - center_y)^2 + (j - center_x)^2); %calculates the distance of the pixel (k,l) from the center of the
apple
            if r > rmax
                ICa(i,j) = 0;
            end
        end
    end

    for i = 1:rows
        for j = 1:columns

            r = sqrt((i - center_y)^2 + (j - center_x)^2); %calculates the distance of the pixel (i,j) from the center of the
well

            if r <= r1
                R1 = R1 + 1;
                if ICa(i,j) == 1
                    R1_count(f,1) = R1_count(f,1) + 1;
                end
            end

            if r > r1 && r <= r2
                R2 = R2 + 1;
                if ICa(i,j) == 1
                    R2_count(f,1) = R2_count(f,1) + 1;
                end
            end
        end
    end
end

```

```

    if r > r2 && r <= r3
        R3 = R3 + 1;
        if ICa(i,j) == 1
            R3_count(f,1) = R3_count(f,1) + 1;
        end
    end

    if r > r3 && r <= r4
        R4 = R4 + 1;
        if ICa(i,j) == 1
            R4_count(f,1) = R4_count(f,1) + 1;
        end
    end
    % {
    if r > r4 && r <= r5
        R5 = R5 + 1;
        if ICa(i,j) == 1
            R5_count(f,1) = R5_count(f,1) + 1;
        end
    end

    if r > r5 && r <= r6
        R6 = R6 + 1;
        if ICa(i,j) == 1
            R6_count(f,1) = R6_count(f,1) + 1;
        end
    end
    % }
end

new_file_name = strcat('Cropped_BW_',file_name{f});
imwrite(ICa,new_file_name,'tif','Compression','none');

F1_count(f,1) = R1_count(f,1)/R1;
F2_count(f,1) = R2_count(f,1)/R2;
F3_count(f,1) = R3_count(f,1)/R3;
F4_count(f,1) = R4_count(f,1)/R4;
%F5_count(f,1) = R5_count(f,1)/R5;
%F6_count(f,1) = R6_count(f,1)/R6;

% {
BW_s = edge(B,'sobel',0.05);
se = strel('diamond',2);
BW_sdil = imclose(BW_s, se);
BW_dfill = imfill(BW_sdil,'holes');
BW_B = bwareaopen(BW_dfill,75);

CC = bwconncomp(BW_B);
nuclei(i) = CC.NumObjects;
% }

ICB = imcrop(B,RECT);
BW_s = edge(ICB,'sobel',0.05);
se = strel('diamond',2);

```

```

BWsdil = imclose(BWs, se);
BWdfill = imfill(BWsdil,'holes');
BW_B = bwareaopen(BWdfill,75);

B1 = BW_B; B2 = BW_B; B3 = BW_B; B4 = BW_B; B5 = BW_B; B6 = BW_B;
for k = 1:rows
    for l = 1:columns
        r = sqrt((k - center_y)^2 + (l - center_x)^2); %calculates the distance of the pixel (k,l) from the center of the
apple
        if r >= r1          %screens and analyzes region 1
            B1(k,l) = 0;
        end
        if r >= r2
            B2(k,l) = 0;
        end
        if r >= r3
            B3(k,l) = 0;
        end
        if r >= r4
            B4(k,l) = 0;
        end
        % {
        if r >= r5
            B5(k,l) = 0;
        end
        if r >= r6
            B6(k,l) = 0;
        end
        % }
    end
end

B2 = B2 - B1;
B3 = B3 - B2 - B1;
B4 = B4 - B3 - B2 - B1;
% B5 = B5 - B4 - B3 - B2 - B1;
% B6 = B6 - B5 - B4 - B3 - B2 - B1;

new_file_name = strcat('Nuclei_region_1_',file_name{f});
imwrite(B1,new_file_name,'tif','Compression','none');
new_file_name = strcat('Nuclei_region_2_',file_name{f});
imwrite(B2,new_file_name,'tif','Compression','none');
new_file_name = strcat('Nuclei_region_3_',file_name{f});
imwrite(B3,new_file_name,'tif','Compression','none');
new_file_name = strcat('Nuclei_region_4_',file_name{f});
imwrite(B4,new_file_name,'tif','Compression','none');
%new_file_name = strcat('Nuclei_region_5_',file_name{f});
%imwrite(B5,new_file_name,'tif','Compression','none');
%new_file_name = strcat('Nuclei_region_6_',file_name{f});
%imwrite(B6,new_file_name,'tif','Compression','none');

CC1 = bwconncomp(B1);

```

```

CC2 = bwconncomp(B2);
CC3 = bwconncomp(B3);
CC4 = bwconncomp(B4);
%CC5 = bwconncomp(B5);
%CC6 = bwconncomp(B6);

nuclei_regions(:,f) = [CC1.NumObjects; CC2.NumObjects; CC3.NumObjects; CC4.NumObjects];

end

nuclei_regions = nuclei_regions/ring_area;

save('FractionOfActinInSegmentR1.txt','F1_count','-ascii','-double');
save('FractionOfActinInSegmentR2.txt','F2_count','-ascii','-double');
save('FractionOfActinInSegmentR3.txt','F3_count','-ascii','-double');
save('FractionOfActinInSegmentR4.txt','F4_count','-ascii','-double');
%save('FractionOfActinInSegmentR5.txt','F5_count','-ascii','-double');
%save('FractionOfActinInSegmentR6.txt','F6_count','-ascii','-double');
save('NucleiInRegions.txt','nuclei_regions','-ascii','-double');

```


D-2 Protrusion Dynamics of Cell Pairs

```
%CELL-
%
%This code will require that you have collected all the resultant lengths (r)
% and the average angles for all cells analyzed. It will perform a second order
% analysis on the data.
%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Cell L
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Cell L
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Cell L
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

sprintf('%s','Select LEFT CELL Data')
LEFT = xlsread('C:\Users\Anthony Diaz-Santana\Documents\MATLAB\Position1_Data_cell_pairs.xls',-1);

Uncorrected_cell_center_x_left = LEFT(:,1);
Uncorrected_cell_center_y_left = LEFT(:,2);
Uncorrected_slicee_left = LEFT(:,5);
Uncorrected_slice_left = round(Uncorrected_slicee_left/0.3125);

sprintf('%s','Select REFERENCE Data')
REFERENCE = xlsread('C:\Users\Anthony Diaz-Santana\Documents\MATLAB\Position1_Data_cell_pairs.xls',-
1);

if REFERENCE ~= 0
    Ref_x = REFERENCE(:,1);
    Ref_y = REFERENCE(:,2);
    Ref_slicee = REFERENCE(:,5);
    if Ref_slicee ~= 0
        Ref_slice = round(Ref_slicee/0.3125);
    end
else
    Ref_slicee = 0;
end

sprintf('%s','Select LEFT CELL PROTRUSION Data')
PROTRUSION_LEFT = xlsread('C:\Users\Anthony Diaz-
Santana\Documents\MATLAB\Position1_Data_cell_pairs.xls',-1);

Uncorrected_Protrusions_x_left = PROTRUSION_LEFT(:,1);
Uncorrected_Protrusions_y_left = PROTRUSION_LEFT(:,2);
Protrusions_slicee_left = PROTRUSION_LEFT(:,5);
Protrusions_slice_left = round(Protrusions_slicee_left/0.3125);

interval = input('What is the time interval used (in minutes)? _ ');
cell_area_left = input('What is the area of the LEFT cell in square microns? _ ');
cell_radius_left = sqrt(cell_area_left/pi);
%protrusion_threshold = input('What is the protrusion threshold in microns you want to use? _ ');
```

```

%%%%%%%%%% Cell R
%%%%%%%%%%
%%%%%%%%%% Cell R
%%%%%%%%%%
%%%%%%%%%% Cell R
%%%%%%%%%%

sprintf('%s','Select RIGHT CELL Data')
RIGHT = xlsread('C:\Users\Anthony Diaz-Santana\Documents\MATLAB\Position1_Data_cell_pairs.xls',-1);

Uncorrected_cell_center_x_right = RIGHT(:,1);
Uncorrected_cell_center_y_right = RIGHT(:,2);
Uncorrected_slice_right = RIGHT(:,5);
Uncorrected_slice_right = round(Uncorrected_slice_right/0.3125);

sprintf('%s','Select RIGHT CELL PROTRUSION Data')
PROTRUSION_RIGHT = xlsread('C:\Users\Anthony Diaz-Santana\Documents\MATLAB\Position1_Data_cell_pairs.xls',-1);

Uncorrected_Protrusions_x_right = PROTRUSION_RIGHT(:,1);
Uncorrected_Protrusions_y_right = PROTRUSION_RIGHT(:,2);
Protrusions_slice_right = PROTRUSION_RIGHT(:,5);
Protrusions_slice_right = round(Protrusions_slice_right/0.3125);

cell_area_right = input('What is the area of the RIGHT cell in square microns? ');
cell_radius_right = sqrt(cell_area_right/pi);

initial_separation = sqrt((Uncorrected_cell_center_x_right(1) - Uncorrected_cell_center_x_left(1))^2 +
(Uncorrected_cell_center_y_right(1) - Uncorrected_cell_center_y_left(1))^2);
separation = sqrt((Uncorrected_cell_center_x_right - Uncorrected_cell_center_x_left).^2 +
(Uncorrected_cell_center_y_right - Uncorrected_cell_center_y_left).^2);
average_separation = mean(separation);

%%%%%%%%%%
%%%%%%%%%%
save('Separation with Time.txt', 'separation','-ascii', '-double');
%save('Average_Separation.txt', 'average_separation','-ascii', '-double');

%%%%%%%%%%
%%%%%%%%%%

i = length(Uncorrected_cell_center_x_right);
w_left = length(Uncorrected_Protrusions_y_left)
F_2_w_left_minus_2 = input('What is the F-distribution value for the left cell?: ');
w_right = length(Uncorrected_Protrusions_y_right)
F_2_w_right_minus_2 = input('What is the F-distribution value for the right cell?: ');
t_w_left_minus_1 = input('What is the t-distribution value for the left cell?: ');
t_w_right_minus_1 = input('What is the t-distribution value for the right cell?: ');

delta_Ref_x = zeros(i,1);
delta_Ref_x(1,1) = 0;
delta_Ref_y = zeros(i,1);
delta_Ref_y(1,1) = 0;

```

```

deltaUncorrected_cell_center_x_right = zeros(i,1);
deltaUncorrected_cell_center_y_right = zeros(i,1);
deltaUncorrected_cell_center_x_right(1,1) = 0;
deltaUncorrected_cell_center_y_right(1,1) = 0;

deltaUncorrected_cell_center_x_left = zeros(i,1);
deltaUncorrected_cell_center_y_left = zeros(i,1);
deltaUncorrected_cell_center_x_left(1,1) = 0;
deltaUncorrected_cell_center_y_left(1,1) = 0;

deltaUncorrected_protrusion_x_left = zeros(w_left,1);
deltaUncorrected_protrusion_y_left = zeros(w_left,1);
deltaUncorrected_protrusion_x_left(1,1) = 0;
deltaUncorrected_protrusion_y_left(1,1) = 0;

deltaUncorrected_protrusion_x_right = zeros(w_right,1);
deltaUncorrected_protrusion_y_right = zeros(w_right,1);
deltaUncorrected_protrusion_x_right(1,1) = 0;
deltaUncorrected_protrusion_y_right(1,1) = 0;

time_interval_in_hours = ((1/60)*0:interval:i*interval)';
%time = time_min/60;
time_in_hours = i*interval/60;
Protrusion_Histogram_left = zeros(i,1);
Protrusion_Histogram_right = zeros(i,1);
protrusion_length_left_over_time = zeros(i,1);
protrusion_length_right_over_time = zeros(i,1);
Protrusion_vectors_from_membrane_left_x = zeros(i,1);
Protrusion_vectors_from_membrane_left_y = zeros(i,1);
Protrusion_unit_vectors_left_x = zeros(i,1);
Protrusion_unit_vectors_left_y = zeros(i,1);
Protrusion_vectors_from_membrane_right_x = zeros(i,1);
Protrusion_vectors_from_membrane_right_y = zeros(i,1);
Protrusion_unit_vectors_right_x = zeros(i,1);
Protrusion_unit_vectors_right_y = zeros(i,1);

if Ref_slicee ~= 0
    for j = 2:i
        delta_Ref_x(j,1) = Ref_x(j,1) - Ref_x(j - 1,1);
        delta_Ref_y(j,1) = Ref_y(j,1) - Ref_y(j - 1,1);
        deltaUncorrected_cell_center_x_left(j,1) = Uncorrected_cell_center_x_left(j,1) -
        Uncorrected_cell_center_x_left(j - 1,1);
        deltaUncorrected_cell_center_y_left(j,1) = Uncorrected_cell_center_y_left(j,1) -
        Uncorrected_cell_center_y_left(j - 1,1);
        deltaUncorrected_cell_center_x_right(j,1) = Uncorrected_cell_center_x_right(j,1) -
        Uncorrected_cell_center_x_right(j - 1,1);
        deltaUncorrected_cell_center_y_right(j,1) = Uncorrected_cell_center_y_right(j,1) -
        Uncorrected_cell_center_y_right(j - 1,1);
    end
    c_cell_center_x_left = Uncorrected_cell_center_x_left + (deltaUncorrected_cell_center_x_left - delta_Ref_x);
    c_cell_center_y_left = Uncorrected_cell_center_y_left + (deltaUncorrected_cell_center_y_left - delta_Ref_y);
    c_cell_center_x_right = Uncorrected_cell_center_x_right + (deltaUncorrected_cell_center_x_right -
    delta_Ref_x);
    c_cell_center_y_right = Uncorrected_cell_center_y_right + (deltaUncorrected_cell_center_y_right -
    delta_Ref_y);

```

```

else
    c_cell_center_x_left = Uncorrected_cell_center_x_left;
    c_cell_center_y_left = Uncorrected_cell_center_y_left;
    c_cell_center_x_right = Uncorrected_cell_center_x_right;
    c_cell_center_y_right = Uncorrected_cell_center_y_right;
end
save('Corrected_Cell_Center_x_Left.txt','c_cell_center_x_left','-ascii','-double');
save('Corrected_Cell_Center_y_Left.txt','c_cell_center_y_left','-ascii','-double');
save('Corrected_Cell_Center_x_Right.txt','c_cell_center_x_right','-ascii','-double');
save('Corrected_Cell_Center_y_Right.txt','c_cell_center_y_right','-ascii','-double');

if Ref_slicee ~= 0
    for j = 2:w_left
        deltaUncorrected_protrusion_x_left(j,1) = Uncorrected_Protrusions_x_left(j,1) -
        Uncorrected_Protrusions_x_left(j - 1,1);
        deltaUncorrected_protrusion_y_left(j,1) = Uncorrected_Protrusions_y_left(j,1) -
        Uncorrected_Protrusions_y_left(j - 1,1);
        c_Protrusions_x_left(j,1) = Uncorrected_Protrusions_x_left(j,1) + (deltaUncorrected_protrusion_x_left(j,1) -
        delta_Ref_x(Protrusions_slice_left(j)));
        c_Protrusions_y_left(j,1) = Uncorrected_Protrusions_y_left(j,1) + (deltaUncorrected_protrusion_y_left(j,1) -
        delta_Ref_y(Protrusions_slice_left(j)));
    end
    c_Protrusions_x_left(1,1) = Uncorrected_Protrusions_x_left(1,1) + (deltaUncorrected_protrusion_x_left(1,1) -
    delta_Ref_x(Protrusions_slice_left(1)));
    c_Protrusions_y_left(1,1) = Uncorrected_Protrusions_y_left(1,1) + (deltaUncorrected_protrusion_y_left(1,1) -
    delta_Ref_y(Protrusions_slice_left(1)));
else
    c_Protrusions_x_left = Uncorrected_Protrusions_x_left;
    c_Protrusions_y_left = Uncorrected_Protrusions_y_left;
end

if Ref_slicee ~= 0
    for j = 2:w_right
        deltaUncorrected_protrusion_x_right(j,1) = Uncorrected_Protrusions_x_right(j,1) -
        Uncorrected_Protrusions_x_right(j - 1,1);
        deltaUncorrected_protrusion_y_right(j,1) = Uncorrected_Protrusions_y_right(j,1) -
        Uncorrected_Protrusions_y_right(j - 1,1);
        c_Protrusions_x_right(j,1) = Uncorrected_Protrusions_x_right(j,1) +
        (deltaUncorrected_protrusion_x_right(j,1) - delta_Ref_x(Protrusions_slice_right(j)));
        c_Protrusions_y_right(j,1) = Uncorrected_Protrusions_y_right(j,1) +
        (deltaUncorrected_protrusion_y_right(j,1) - delta_Ref_y(Protrusions_slice_right(j)));
    end
    c_Protrusions_x_right(1,1) = Uncorrected_Protrusions_x_right(1,1) +
    (deltaUncorrected_protrusion_x_right(1,1) - delta_Ref_x(Protrusions_slice_right(1)));
    c_Protrusions_y_right(1,1) = Uncorrected_Protrusions_y_right(1,1) +
    (deltaUncorrected_protrusion_y_right(1,1) - delta_Ref_y(Protrusions_slice_right(1)));
else
    c_Protrusions_x_right = Uncorrected_Protrusions_x_right;
    c_Protrusions_y_right = Uncorrected_Protrusions_y_right;
end

Phi = atan((c_cell_center_y_right - c_cell_center_y_left)/(c_cell_center_x_right - c_cell_center_x_left)); %%
angle between two cells for every time point

```

%%%%%%%% This section assigns each protrusion to its time point,

```

%%%%%%%%% correlating it with its appropriate cell center coordinate. It
%%%%%%%%% also transforms the coordinate systems to the set that contains
%%%%%%%%% the "x-axis" aligned with the two cells (which is dynamical
%%%%%%%%% because the two cells are changing their locations)%%%%%%%%%

for m = 1:w_left
    x_P_double_prime_left(m,1) = (c_Protrusions_x_left(m,1) -
c_cell_center_x_left(Protrusions_slice_left(m)))*cos(Phi(Protrusions_slice_left(m))) + (c_Protrusions_y_left(m,1) -
c_cell_center_y_left(Protrusions_slice_left(m)))*sin(Phi(Protrusions_slice_left(m)));
    y_P_double_prime_left(m,1) = -(c_Protrusions_x_left(m,1) -
c_cell_center_x_left(Protrusions_slice_left(m)))*sin(Phi(Protrusions_slice_left(m))) + (c_Protrusions_y_left(m,1) -
c_cell_center_y_left(Protrusions_slice_left(m)))*cos(Phi(Protrusions_slice_left(m)));
    for q = 1:i
        if q == Protrusions_slice_left(m)
            Protrusion_Histogram_left(q,1) = Protrusion_Histogram_left(q,1) + 1;
        end
    end
end
save('Protrusion_Histogram_with_Time_Left.txt','Protrusion_Histogram_left','-ascii','-double');

% Flips the entire coordinates about the y-axis so that the angles calculated agree with the 0 degrees direction
being towards the other cell always.
c_Protrusions_x_right = -c_Protrusions_x_right;
c_cell_center_x_right = -c_cell_center_x_right;
c_cell_center_x_left = -c_cell_center_x_left;

Phi = atan((c_cell_center_y_right - c_cell_center_y_left)./(c_cell_center_x_right - c_cell_center_x_left)); %%
angle between two cells for every time point

for m = 1:w_right
    x_P_double_prime_right(m,1) = (c_Protrusions_x_right(m,1) -
c_cell_center_x_right(Protrusions_slice_right(m)))*cos(Phi(Protrusions_slice_right(m))) +
(c_Protrusions_y_right(m,1) -
c_cell_center_y_right(Protrusions_slice_right(m)))*sin(Phi(Protrusions_slice_right(m)));
    y_P_double_prime_right(m,1) = -(c_Protrusions_x_right(m,1) -
c_cell_center_x_right(Protrusions_slice_right(m)))*sin(Phi(Protrusions_slice_right(m))) +
(c_Protrusions_y_right(m,1) -
c_cell_center_y_right(Protrusions_slice_right(m)))*cos(Phi(Protrusions_slice_right(m)));
    for q = 1:i
        if q == Protrusions_slice_right(m)
            Protrusion_Histogram_right(q,1) = Protrusion_Histogram_right(q,1) + 1;
        end
    end
end
save('Protrusion_Histogram_with_Time_Right .txt','Protrusion_Histogram_right','-ascii','-double');
save('time_in_hours.txt','time_interval_in_hours','-ascii','-double');

```

```

%%%%%%%%%
%%%%%%%%%
%%%%%%%%%

```

```

%%%%%%%%% Calculating angle of protrusion in the appropriate coordinate
%%%%%%%%% system and correcting for quadrant location

```

```

Protrusion_angle_left = atan(y_P_double_prime_left./x_P_double_prime_left);
Protrusions_Unit_Circle_x_left = cos(Protrusion_angle_left);
Protrusions_Unit_Circle_y_left = sin(Protrusion_angle_left);

```

```

for m = 1:w_left

```

```

    if x_P_double_prime_left(m,1) < 0
        Protrusion_angle_left(m,1) = Protrusion_angle_left(m,1) + pi;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
    if x_P_double_prime_left(m,1) > 0 && y_P_double_prime_left(m,1) < 0
        Protrusion_angle_left(m,1) = Protrusion_angle_left(m,1) + 2*pi;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
    if x_P_double_prime_left(m,1) == 0 && y_P_double_prime_left(m,1) < 0
        Protrusion_angle_left(m,1) = 3*pi/2;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
    if x_P_double_prime_left(m,1) == 0 && y_P_double_prime_left(m,1) > 0
        Protrusion_angle_left(m,1) = pi/2;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
end
circ_plot(Protrusion_angle_left);
title('Left Cell');

```

```

Protrusion_angle_left
%%%%%%%%%%%%%%

```

```

Protrusion_angle_right = atan(y_P_double_prime_right./x_P_double_prime_right);
Protrusions_Unit_Circle_x_right = cos(Protrusion_angle_right);
Protrusions_Unit_Circle_y_right = sin(Protrusion_angle_right);

```

```

for m = 1:w_right

```

```

    if x_P_double_prime_right(m,1) < 0
        Protrusion_angle_right(m,1) = Protrusion_angle_right(m,1) + pi;
        Protrusions_Unit_Circle_x_right(m,1) = cos(Protrusion_angle_right(m,1));
        Protrusions_Unit_Circle_y_right(m,1) = sin(Protrusion_angle_right(m,1));
    end
    if x_P_double_prime_right(m,1) > 0 && y_P_double_prime_right(m,1) < 0
        Protrusion_angle_right(m,1) = Protrusion_angle_right(m,1) + 2*pi;
        Protrusions_Unit_Circle_x_right(m,1) = cos(Protrusion_angle_right(m,1));
        Protrusions_Unit_Circle_y_right(m,1) = sin(Protrusion_angle_right(m,1));
    end
    if x_P_double_prime_right(m,1) == 0 && y_P_double_prime_right(m,1) < 0
        Protrusion_angle_right(m,1) = 3*pi/2;
        Protrusions_Unit_Circle_x_right(m,1) = cos(Protrusion_angle_right(m,1));
        Protrusions_Unit_Circle_y_right(m,1) = sin(Protrusion_angle_right(m,1));
    end
end

```

```

    if x_P_double_prime_right(m,1) == 0 && y_P_double_prime_right(m,1) > 0
        Protrusion_angle_right(m,1) = pi/2;
        Protrusions_Unit_Circle_x_right(m,1) = cos(Protrusion_angle_right(m,1));
        Protrusions_Unit_Circle_y_right(m,1) = sin(Protrusion_angle_right(m,1));
    end
end
figure()
circ_plot(Protrusion_angle_right);
title('Right Cell');

%%%%%%%%%%%%%%

figure()

persistence_plot_time_left = Protrusions_slice_left*interval/60;
plot(persistence_plot_time_left, Protrusion_angle_left*180/pi,'or')
save('Time of Protrusions for Persistence Plot LEFT.txt','persistence_plot_time_left','-ascii','-double');

persistence_plot_time_right = Protrusions_slice_right*interval/60;
plot(persistence_plot_time_right, Protrusion_angle_right*180/pi,'or')
save('Time of Protrusions for Persistence Plot RIGHT.txt','persistence_plot_time_right','-ascii','-double');

x_UnitCircle_left = Protrusions_Unit_Circle_x_left;
y_UnitCircle_left = Protrusions_Unit_Circle_y_left;
x_UnitCircle_right = Protrusions_Unit_Circle_x_right;
y_UnitCircle_right = Protrusions_Unit_Circle_y_right;

y_UnitCircle_avg_left = mean(y_UnitCircle_left);
x_UnitCircle_avg_left = mean(x_UnitCircle_left);
y_UnitCircle_avg_right = mean(y_UnitCircle_right);
x_UnitCircle_avg_right = mean(x_UnitCircle_right);

resultant_length_UnitCircle_left = sqrt(x_UnitCircle_avg_left^2 + y_UnitCircle_avg_left^2);
resultant_length_UnitCircle_right = sqrt(x_UnitCircle_avg_right^2 + y_UnitCircle_avg_right^2);

std_UnitCircle_x_left = std(x_UnitCircle_left);
std_UnitCircle_y_left = std(y_UnitCircle_left);
std_UnitCircle_x_right = std(x_UnitCircle_right);
std_UnitCircle_y_right = std(y_UnitCircle_right);

Theta_Ave_left = atan(y_UnitCircle_avg_left/x_UnitCircle_avg_left);
if x_UnitCircle_avg_left < 0
    Theta_Ave_left = Theta_Ave_left + pi;
end
if x_UnitCircle_avg_left > 0 && y_UnitCircle_avg_left < 0
    Theta_Ave_left = Theta_Ave_left + 2*pi;
end
if x_UnitCircle_avg_left == 0 && y_UnitCircle_avg_left < 0
    Theta_Ave_left = 3*pi/2;
end
if x_UnitCircle_avg_left == 0 && y_UnitCircle_avg_left > 0
    Theta_Ave_left = pi/2;
end

```

```

Theta_Ave_right = atan(y_UnitCircle_avg_right/x_UnitCircle_avg_right);
if x_UnitCircle_avg_right < 0
    Theta_Ave_right = Theta_Ave_right + pi;
end
if x_UnitCircle_avg_right > 0 && y_UnitCircle_avg_right < 0
    Theta_Ave_right = Theta_Ave_right + 2*pi;
end
if x_UnitCircle_avg_right == 0 && y_UnitCircle_avg_right < 0
    Theta_Ave_right = 3*pi/2;
end
if x_UnitCircle_avg_right == 0 && y_UnitCircle_avg_right > 0
    Theta_Ave_right = pi/2;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% V-Test
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for phi = (1:360)
    V_left(phi) = w_left*resultant_length_UnitCircle_left*cos(Theta_Ave_left - phi*pi/180);
    V_right(phi) = w_right*resultant_length_UnitCircle_right*cos(Theta_Ave_right - phi*pi/180);
end
u_left = V_left*sqrt(2/w_left);
u_right = V_right*sqrt(2/w_right);
u_left = u_left';
u_right = u_right';

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Rao's Spacing Test %%%%%%%%%
Theta_Ave_left = Theta_Ave_left*180/pi;
Protrusion_angle_left
Protrusion_angle_left = Protrusion_angle_left*180/pi;
Protrusion_angle_left
save('Angle of Protrusions LEFT.txt','Protrusion_angle_left','-ascii','-double');

lambda_left = 360/w_left;    %degrees hereon
sorted_angles_left = sort(Protrusion_angle_left);
T_left(1) = sorted_angles_left(2) - sorted_angles_left(1);
for a = 2:(w_left - 1)
    T_left(a) = sorted_angles_left(a + 1) - sorted_angles_left(a);
end
T_left = T_left';
U_argument_left = abs(T_left - lambda_left);
U_left = 1/2*sum(U_argument_left);
Protrusion_angle_left

Theta_Ave_right = Theta_Ave_right*180/pi;
Protrusion_angle_right = Protrusion_angle_right*180/pi;
save('Angle of Protrusions RIGHT.txt','Protrusion_angle_right','-ascii','-double');

lambda_right = 360/w_right;    %degrees hereon
sorted_angles_right = sort(Protrusion_angle_right);
T_right(1) = sorted_angles_right(2) - sorted_angles_right(1);
for a = 2:(w_right - 1)
    T_right(a) = sorted_angles_right(a + 1) - sorted_angles_right(a);
end
T_right = T_right';
U_argument_right = abs(T_right - lambda_right);

```



```

U_right = 1/2*sum(U_argument_right);
Protrusion_angle_right

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Rayleigh's Test %%%%%%%%%
z_left = w_left*resultant_length_UnitCircle_left^2;
z_right = w_right*resultant_length_UnitCircle_right^2;

P_left = exp(sqrt(1 + 4*w_left + 4*(w_left^2 - (w_left*resultant_length_UnitCircle_left)^2)) - (1 + 2*w_left));
P_right = exp(sqrt(1 + 4*w_right + 4*(w_right^2 - (w_right*resultant_length_UnitCircle_right)^2)) - (1 + 2*w_right));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Confidence Interval for Average Angles on Unit Circle %%%%%%%%%
chi_square = 3.841; %95 percent confidence_interval
if resultant_length_UnitCircle_left <= 0.9 && resultant_length_UnitCircle_left > sqrt(chi_square/(2*w_left))
    d_unit_circle_left = ceil((180/pi)*acos(sqrt((2*w_left*(2*(w_left*resultant_length_UnitCircle_left)^2 - w_left*chi_square))/(4*w_left - chi_square)))/w_left*resultant_length_UnitCircle_left));
    save('Confidence_Interval_UnitCircle_left.txt','d_unit_circle_left','-ascii','-double');
end
if resultant_length_UnitCircle_right <= 0.9 && resultant_length_UnitCircle_right > sqrt(chi_square/(2*w_right))
    d_unit_circle_right = ceil((180/pi)*acos(sqrt((2*w_right*(2*(w_right*resultant_length_UnitCircle_right)^2 - w_right*chi_square))/(4*w_right - chi_square)))/w_right*resultant_length_UnitCircle_right));
    save('Confidence_Interval_UnitCircle_right.txt','d_unit_circle_right','-ascii','-double');
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

x_confidence_interval_UnitCircle_left = t_w_left_minus_1*std_UnitCircle_x_left/sqrt(w_left);
y_confidence_interval_UnitCircle_left = t_w_left_minus_1*std_UnitCircle_y_left/sqrt(w_left);
x_confidence_interval_UnitCircle_right = t_w_right_minus_1*std_UnitCircle_x_right/sqrt(w_right);
y_confidence_interval_UnitCircle_right = t_w_right_minus_1*std_UnitCircle_y_right/sqrt(w_right);

summary_UnitCircle_left = [initial_separation Theta_Ave_left resultant_length_UnitCircle_left w_left P_left U_left
x_UnitCircle_avg_left std_UnitCircle_x_left x_confidence_interval_UnitCircle_left y_UnitCircle_avg_left
std_UnitCircle_y_left y_confidence_interval_UnitCircle_left average_separation]';
summary_UnitCircle_right = [initial_separation Theta_Ave_right resultant_length_UnitCircle_right w_right P_right
U_right x_UnitCircle_avg_right std_UnitCircle_x_right x_confidence_interval_UnitCircle_right
y_UnitCircle_avg_right std_UnitCircle_y_right y_confidence_interval_UnitCircle_right average_separation]';

save('Summary_UnitCircle_Left.txt','summary_UnitCircle_left','-ascii','-double');
save('Summary_UnitCircle_Right.txt','summary_UnitCircle_right','-ascii','-double');
save('Critical_Value_for_V-Test(u)_Left.txt','u_left','-ascii','-double');
save('Critical_Value_for_V-Test(u)_Right.txt','u_right','-ascii','-double');
%save('x-coordinates_protrusions_left.txt','x_UnitCircle_left','-ascii','-double');
%save('y-coordinates_protrusions_left.txt','y_UnitCircle_left','-ascii','-double');
%save('x-coordinates_protrusions_right.txt','x_UnitCircle_right','-ascii','-double');
%save('y-coordinates_protrusions_right.txt','y_UnitCircle_right','-ascii','-double');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Protrusion_angle_left = Protrusion_angle_left*pi/180;
Protrusion_angle_right = Protrusion_angle_right*pi/180;

x_left = x_P_double_prime_left - cell_radius_left*cos(Protrusion_angle_left);

```

```

y_left = y_P_double_prime_left - cell_radius_left*sin(Protrusion_angle_left);
x_right = x_P_double_prime_right - cell_radius_right*cos(Protrusion_angle_right);
y_right = y_P_double_prime_right - cell_radius_right*sin(Protrusion_angle_right);

save('Protrusion_Length_from_Membrane_x_left.txt','x_left','-ascii','-double');
save('Protrusion_Length_from_Membrane_y_left.txt','y_left','-ascii','-double');
save('Protrusion_Length_from_Membrane_x_right.txt','x_right','-ascii','-double');
save('Protrusion_Length_from_Membrane_y_right.txt','y_right','-ascii','-double');

y_avg_left = mean(y_left);
x_avg_left = mean(x_left);
y_avg_right = mean(y_right);
x_avg_right = mean(x_right);

std_x_left = std(x_left);
std_y_left = std(y_left);
std_x_right = std(x_right);
std_y_right = std(y_right);

s1_squared_left = sum((x_left - x_avg_left).^2*(1/(w_left - 1))); % [length^2]
s2_squared_left = sum((y_left - y_avg_left).^2*(1/(w_left - 1))); % [length^2]
s1_squared_right = sum((x_right - x_avg_right).^2*(1/(w_right - 1))); % [length^2]
s2_squared_right = sum((y_right - y_avg_right).^2*(1/(w_right - 1))); % [length^2]

r_left = sum((x_left - x_avg_left).*(y_left - y_avg_left))/((w_left - 1)*sqrt(s1_squared_left)*sqrt(s2_squared_left));
% [dimensionless]
T_squared_confidence_left = 2*((w_left - 1)/(w_left - 2))*F_2_w_left_minus_2;
T_squared_left = (w_left/(1 - r_left^2))*(x_avg_left^2/s1_squared_left -
2*r_left*x_avg_left*y_avg_left/(sqrt(s1_squared_left)*sqrt(s2_squared_left)) + y_avg_left^2/s2_squared_left);
r_right = sum((x_right - x_avg_right).*(y_right - y_avg_right))/((w_right -
1)*sqrt(s1_squared_right)*sqrt(s2_squared_right)); % [dimensionless]
T_squared_confidence_right = 2*((w_right - 1)/(w_right - 2))*F_2_w_right_minus_2;
T_squared_right = (w_right/(1 - r_right^2))*(x_avg_right^2/s1_squared_right -
2*r_right*x_avg_right*y_avg_right/(sqrt(s1_squared_right)*sqrt(s2_squared_right)) +
y_avg_right^2/s2_squared_right);

A_left = s2_squared_left; % [length^2]
A_right = s2_squared_right; % [length^2]
B_left = -r_left*sqrt(s1_squared_left)*sqrt(s2_squared_left); % [length^2]
B_right = -r_right*sqrt(s1_squared_right)*sqrt(s2_squared_right); % [length^2]
C_left = s1_squared_left; % [length^2]
C_right = s1_squared_right; % [length^2]
D_left = (1 - r_left^2)*s1_squared_left*s2_squared_left; % [length^4]
D_right = (1 - r_right^2)*s1_squared_right*s2_squared_right; % [length^4]
R_left = sqrt((A_left - C_left)^2 + 4*B_left^2); % [length^2]
R_right = sqrt((A_right - C_right)^2 + 4*B_right^2); % [length^2]
a_left = sqrt(2*D_left/(A_left + C_left - R_left)); % [length]
a_right = sqrt(2*D_right/(A_right + C_right - R_right)); % [length]
b_left = sqrt(2*D_left/(A_left + C_left + R_left)); % [length]
b_right = sqrt(2*D_right/(A_right + C_right + R_right)); % [length]
theta_left = atan(2*B_left/(A_left - C_left - R_left));
theta_right = atan(2*B_right/(A_right - C_right - R_right));

H_left = A_left*C_left - B_left^2;
H_right = A_right*C_right - B_right^2;

```

```

G_left = A_left*x_avg_left^2 + 2*B_left*x_avg_left*y_avg_left + C_left*y_avg_left^2 - D_left;
G_right = A_right*x_avg_right^2 + 2*B_right*x_avg_right*y_avg_right + C_right*y_avg_right^2 - D_right;
U_left = (H_left*x_avg_left^2 - C_left*D_left)^(-1);
U_right = (H_right*x_avg_right^2 - C_right*D_right)^(-1);
V_left = (D_left*G_left*H_left)^(1/2);
V_right = (D_right*G_right*H_right)^(1/2);
W_left = H_left*x_avg_left*y_avg_left + B_left*D_left;
W_right = H_right*x_avg_right*y_avg_right + B_right*D_right;
m1_left = U_left*(W_left + V_left);
m2_left = U_left*(W_left - V_left);
m1_right = U_right*(W_right + V_right);
m2_right = U_right*(W_right - V_right);
m1_theta_left = (180/pi)*atan(m1_left);
m2_theta_left = (180/pi)*atan(m2_left);
m1_theta_right = (180/pi)*atan(m1_right);
m2_theta_right = (180/pi)*atan(m2_right);

D_confidence_left = (1 - r_left^2)*s1_squared_left*s2_squared_left*(1/w_left)*T_squared_confidence_left;
% [length^4]
a_confidence_left = a_left*(w_left^(-1/2))*sqrt(T_squared_confidence_left);
b_confidence_left = b_left*(w_left^(-1/2))*sqrt(T_squared_confidence_left);
D_confidence_right = (1 -
r_right^2)*s1_squared_right*s2_squared_right*(1/w_right)*T_squared_confidence_right; % [length^4]
a_confidence_right = a_right*(w_right^(-1/2))*sqrt(T_squared_confidence_right);
b_confidence_right = b_right*(w_right^(-1/2))*sqrt(T_squared_confidence_right);

phi = (1:360)';
phi = phi*(pi/180);
X_left = x_avg_left + a_left*cos(theta_left)*cos(phi) - b_left*sin(theta_left)*sin(phi);
Y_left = y_avg_left + a_left*sin(theta_left)*cos(phi) + b_left*cos(theta_left)*sin(phi);
X_confidence_left = x_avg_left + a_confidence_left*cos(theta_left)*cos(phi) -
b_confidence_left*sin(theta_left)*sin(phi);
Y_confidence_left = y_avg_left + a_confidence_left*sin(theta_left)*cos(phi) +
b_confidence_left*cos(theta_left)*sin(phi);
X_right = x_avg_right + a_right*cos(theta_right)*cos(phi) - b_right*sin(theta_right)*sin(phi);
Y_right = y_avg_right + a_right*sin(theta_right)*cos(phi) + b_right*cos(theta_right)*sin(phi);
X_confidence_right = x_avg_right + a_confidence_right*cos(theta_right)*cos(phi) -
b_confidence_right*sin(theta_right)*sin(phi);
Y_confidence_right = y_avg_right + a_confidence_right*sin(theta_right)*cos(phi) +
b_confidence_right*cos(theta_right)*sin(phi);

figure()
plot(X_left,Y_left,'b')
hold on
plot(x_left,y_left,'ro')
plot(X_confidence_left,Y_confidence_left,'k')
hold off

figure()
plot(X_right,Y_right,'b')
hold on
plot(x_right,y_right,'ro')
plot(X_confidence_right,Y_confidence_right,'k')
hold off

```

```

x_confidence_interval_left = t_w_left_minus_1*std_x_left/sqrt(w_left);
y_confidence_interval_left = t_w_left_minus_1*std_y_left/sqrt(w_left);
x_confidence_interval_right = t_w_right_minus_1*std_x_right/sqrt(w_right);
y_confidence_interval_right = t_w_right_minus_1*std_y_right/sqrt(w_right);

theta_left = theta_left*180/pi;
theta_right = theta_right*180/pi;

avg_resultant_vector_angle_left = atan(y_avg_left/x_avg_left);
avg_resultant_vector_angle_right = atan(y_avg_right/x_avg_right);

if x_avg_left < 0
    avg_resultant_vector_angle_left = avg_resultant_vector_angle_left + pi;
end
if x_avg_left > 0 && y_avg_left < 0
    avg_resultant_vector_angle_left = avg_resultant_vector_angle_left + 2*pi;
end
if x_avg_left == 0 && y_avg_left < 0
    avg_resultant_vector_angle_left = 3*pi/2;
end
if x_avg_left == 0 && y_avg_left > 0
    avg_resultant_vector_angle_left = pi/2;
end

if x_avg_right < 0
    avg_resultant_vector_angle_right = avg_resultant_vector_angle_right + pi;
end
if x_avg_right > 0 && y_avg_right < 0
    avg_resultant_vector_angle_right = avg_resultant_vector_angle_right + 2*pi;
end
if x_avg_right == 0 && y_avg_right < 0
    avg_resultant_vector_angle_right = 3*pi/2;
end
if x_avg_right == 0 && y_avg_right > 0
    avg_resultant_vector_angle_right = pi/2;
end

avg_resultant_vector_angle_left = (180/pi)*avg_resultant_vector_angle_left;
avg_resultant_vector_angle_right = (180/pi)*avg_resultant_vector_angle_right;
mean_vector_magnitude_left = mean(sqrt(x_left.^2 + y_left.^2));
mean_vector_magnitude_right = mean(sqrt(x_right.^2 + y_right.^2));
vector_magnitude_left = sqrt(x_left.^2 + y_left.^2);
vector_magnitude_right = sqrt(x_right.^2 + y_right.^2);

for q = 1:i
    for m = 1:w_left
        if Protrusions_slice_left(m) == q
            %vector_magnitude_left(m,1)
            %protrusion_length_left_over_time(q,1)
            protrusion_length_left_over_time(q,1) = protrusion_length_left_over_time(q,1) +
vector_magnitude_left(m,1);
        end
    end
    instantaneous_average_protrusion_length_left(q,1) =
protrusion_length_left_over_time(q,1)/Protrusion_Histogram_left(q);

```

```

end

for q = 1:i
    for m = 1:w_right
        if Protrusions_slice_right(m) == q
            protrusion_length_right_over_time(q,1) = protrusion_length_right_over_time(q,1) +
vector_magnitude_right(m,1);
        end
    end
    instantaneous_average_protrusion_length_right(q,1) =
protrusion_length_right_over_time(q,1)/Protrusion_Histogram_right(q);
end

protrusion_frequency_left = w_left/time_in_hours;
protrusion_frequency_right = w_right/time_in_hours;

summary_left = [initial_separation theta_left avg_resultant_vector_angle_left T_squared_left
T_squared_confidence_left a_left w_left m1_theta_left m2_theta_left mean_vector_magnitude_left x_avg_left
std_x_left x_confidence_interval_left y_avg_left std_y_left y_confidence_interval_left protrusion_frequency_left];
summary_right = [initial_separation theta_right avg_resultant_vector_angle_right T_squared_right
T_squared_confidence_right a_right w_right m1_theta_right m2_theta_right mean_vector_magnitude_right
x_avg_right std_x_right x_confidence_interval_right y_avg_right std_y_right y_confidence_interval_right
protrusion_frequency_right];

save('x-coordinates_confidence_ellipse_LEFT.txt','X_confidence_left','-ascii','-double');
save('y-coordinates_confidence_ellipse_LEFT.txt','Y_confidence_left','-ascii','-double');
save('x-coordinates_confidence_ellipse_RIGHT.txt','X_confidence_right','-ascii','-double');
save('y-coordinates_confidence_ellipse_RIGHT.txt','Y_confidence_right','-ascii','-double');
save('Summary Left.txt','summary_left','-ascii','-double');
save('Summary Right.txt','summary_right','-ascii','-double');
save('Protrusion Length Over Time_LEFT.txt','instantaneous_average_protrusion_length_left','-ascii','-double');
save('Protrusion Length Over Time_RIGHT.txt','instantaneous_average_protrusion_length_right','-ascii','-double');
%save('x-coordinates_protrusions_left.txt','x_UnitCircle_left','-ascii','-double');
%save('y-coordinates_protrusions_left.txt','y_UnitCircle_left','-ascii','-double');
%save('x-coordinates_protrusions_right.txt','x_UnitCircle_right','-ascii','-double');
%save('y-coordinates_protrusions_right.txt','y_UnitCircle_right','-ascii','-double');

%%%%%%%%%% Binning %%%%%%%%%%

%BIN_10_left = hist(Protrusion_angle_left,36);
%BIN_10_left = BIN_10_left';

Binned_protrusions_30_left = zeros(12,1);
Binned_protrusions_180_left = zeros(2,1);
Binned_protrusions_30_right = zeros(12,1);
Binned_protrusions_180_right = zeros(2,1);

for m = 1:w_left
    if 0 <= Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= pi/6
        Binned_protrusions_30_left(1) = Binned_protrusions_30_left(1) + 1;
        Binned_protrusions_180_left(1) = Binned_protrusions_180_left(1) + 1;
    end

    if pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= pi/3
        Binned_protrusions_30_left(2) = Binned_protrusions_30_left(2) + 1;

```

```

    Binned_protrusions_180_left(1) = Binned_protrusions_180_left(1) + 1;
end

if pi/3 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= pi/2
    Binned_protrusions_30_left(3) = Binned_protrusions_30_left(3) + 1;
    Binned_protrusions_180_left(1) = Binned_protrusions_180_left(1) + 1;
end

if pi/2 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 2*pi/3
    Binned_protrusions_30_left(4) = Binned_protrusions_30_left(4) + 1;
    Binned_protrusions_180_left(2) = Binned_protrusions_180_left(2) + 1;
end

if 2*pi/3 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 5*pi/6
    Binned_protrusions_30_left(5) = Binned_protrusions_30_left(5) + 1;
    Binned_protrusions_180_left(2) = Binned_protrusions_180_left(2) + 1;
end

if 5*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= pi
    Binned_protrusions_30_left(6) = Binned_protrusions_30_left(6) + 1;
    Binned_protrusions_180_left(2) = Binned_protrusions_180_left(2) + 1;
end

if pi < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 7*pi/6
    Binned_protrusions_30_left(7) = Binned_protrusions_30_left(7) + 1;
    Binned_protrusions_180_left(2) = Binned_protrusions_180_left(2) + 1;
end

if 7*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 8*pi/6
    Binned_protrusions_30_left(8) = Binned_protrusions_30_left(8) + 1;
    Binned_protrusions_180_left(2) = Binned_protrusions_180_left(2) + 1;
end

if 8*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 9*pi/6
    Binned_protrusions_30_left(9) = Binned_protrusions_30_left(9) + 1;
    Binned_protrusions_180_left(2) = Binned_protrusions_180_left(2) + 1;
end

if 9*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 10*pi/6
    Binned_protrusions_30_left(10) = Binned_protrusions_30_left(10) + 1;
    Binned_protrusions_180_left(1) = Binned_protrusions_180_left(1) + 1;
end

if 10*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 11*pi/6
    Binned_protrusions_30_left(11) = Binned_protrusions_30_left(11) + 1;
    Binned_protrusions_180_left(1) = Binned_protrusions_180_left(1) + 1;
end

if 11*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 2*pi
    Binned_protrusions_30_left(12) = Binned_protrusions_30_left(12) + 1;
    Binned_protrusions_180_left(1) = Binned_protrusions_180_left(1) + 1;
end
end

```

```

for m = 1:w_right
    if 0 <= Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= pi/6
        Binned_protrusions_30_right(1) = Binned_protrusions_30_right(1) + 1;
        Binned_protrusions_180_right(1) = Binned_protrusions_180_right(1) + 1;
    end

    if pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= pi/3
        Binned_protrusions_30_right(2) = Binned_protrusions_30_right(2) + 1;
        Binned_protrusions_180_right(1) = Binned_protrusions_180_right(1) + 1;
    end

    if pi/3 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= pi/2
        Binned_protrusions_30_right(3) = Binned_protrusions_30_right(3) + 1;
        Binned_protrusions_180_right(1) = Binned_protrusions_180_right(1) + 1;
    end

    if pi/2 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 2*pi/3
        Binned_protrusions_30_right(4) = Binned_protrusions_30_right(4) + 1;
        Binned_protrusions_180_right(2) = Binned_protrusions_180_right(2) + 1;
    end

    if 2*pi/3 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 5*pi/6
        Binned_protrusions_30_right(5) = Binned_protrusions_30_right(5) + 1;
        Binned_protrusions_180_right(2) = Binned_protrusions_180_right(2) + 1;
    end

    if 5*pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= pi
        Binned_protrusions_30_right(6) = Binned_protrusions_30_right(6) + 1;
        Binned_protrusions_180_right(2) = Binned_protrusions_180_right(2) + 1;
    end

    if pi < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 7*pi/6
        Binned_protrusions_30_right(7) = Binned_protrusions_30_right(7) + 1;
        Binned_protrusions_180_right(2) = Binned_protrusions_180_right(2) + 1;
    end

    if 7*pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 8*pi/6
        Binned_protrusions_30_right(8) = Binned_protrusions_30_right(8) + 1;
        Binned_protrusions_180_right(2) = Binned_protrusions_180_right(2) + 1;
    end

    if 8*pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 9*pi/6
        Binned_protrusions_30_right(9) = Binned_protrusions_30_right(9) + 1;
        Binned_protrusions_180_right(2) = Binned_protrusions_180_right(2) + 1;
    end

    if 9*pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 10*pi/6
        Binned_protrusions_30_right(10) = Binned_protrusions_30_right(10) + 1;
        Binned_protrusions_180_right(1) = Binned_protrusions_180_right(1) + 1;
    end

    if 10*pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 11*pi/6
        Binned_protrusions_30_right(11) = Binned_protrusions_30_right(11) + 1;
        Binned_protrusions_180_right(1) = Binned_protrusions_180_right(1) + 1;
    end
end

```

```

        if 11*pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 2*pi
            Binned_protrusions_30_right(12) = Binned_protrusions_30_right(12) + 1;
            Binned_protrusions_180_right(1) = Binned_protrusions_180_right(1) + 1;
        end
    end

save('Protrusion_Histogram_Bin30_Left.txt','Binned_protrusions_30_left','-ascii','-double');
save('Protrusion_Histogram_Bin180_Left.txt','Binned_protrusions_180_left','-ascii','-double');
save('Protrusion_Histogram_Bin30_Right.txt','Binned_protrusions_30_right','-ascii','-double');
save('Protrusion_Histogram_Bin180_Right.txt','Binned_protrusions_180_right','-ascii','-double');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Square displacement from Origin %%%%%%%%%%%%%%
MSD_from-Origin_left = (c_cell_center_x_left(:,1) - c_cell_center_x_left(1,1)).^2 + (c_cell_center_y_left(:,1) -
c_cell_center_y_left(1,1)).^2;
MSD_from-Origin_right = (c_cell_center_x_right(:,1) - c_cell_center_x_right(1,1)).^2 + (c_cell_center_y_right(:,1)
- c_cell_center_y_right(1,1)).^2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Squared Displacement from Previous Time position %%%%%%%%%%%%%%
for count = 2:i
    MSD_left(count,1) = (c_cell_center_x_left(count,1) - c_cell_center_x_left(count - 1,1)).^2 +
(c_cell_center_y_left(count,1) - c_cell_center_y_left(count - 1,1)).^2;
    MSD_right(count,1) = (c_cell_center_x_right(count,1) - c_cell_center_x_right(count - 1,1)).^2 +
(c_cell_center_y_right(count,1) - c_cell_center_y_right(count - 1,1)).^2;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

save('Square_Displacement_from-Origin_Left.txt','MSD_from-Origin_left','-ascii','-double');
save('Square_Displacement_from-Origin_Right.txt','MSD_from-Origin_right','-ascii','-double');
save('Square_Displacement_Left.txt','MSD_left','-ascii','-double');
save('Square_Displacement_Right.txt','MSD_right','-ascii','-double');

save('x-Coordinate_Protrusion_Lenghts_Left.txt','x_P_double_prime_left','-ascii','-double');
save('y-Coordinate_Protrusion_Lenghts_Left.txt','y_P_double_prime_left','-ascii','-double');
save('x-Coordinate_Protrusion_Lenghts_Right.txt','x_P_double_prime_right','-ascii','-double');
save('y-Coordinate_Protrusion_Lenghts_Right.txt','y_P_double_prime_right','-ascii','-double');

Phi = (Phi*180/pi)';
save('Angle_Between_Cells_with_Time.txt','Phi','-ascii','-double');

for q = 1:i
    for m = 1:w_left
        if Protrusions_slice_left(m) == q
            Protrusion_vectors_from_membrane_left_x(q,1) = Protrusion_vectors_from_membrane_left_x(q,1) +
x_left(m,1);
            Protrusion_vectors_from_membrane_left_y(q,1) = Protrusion_vectors_from_membrane_left_y(q,1) +
y_left(m,1);
            Protrusion_unit_vectors_left_x(q,1) = Protrusion_unit_vectors_left_x(q,1) + x_UnitCircle_left(m,1);
            Protrusion_unit_vectors_left_y(q,1) = Protrusion_unit_vectors_left_y(q,1) + y_UnitCircle_left(m,1);
        end
    end

    if Protrusion_Histogram_left(q) > 0

```



```

Instantaneous_Average_Vector_left_x(q,1) =
Protrusion_vectors_from_membrane_left_x(q,1)/Protrusion_Histogram_left(q);
Instantaneous_Average_Vector_left_y(q,1) =
Protrusion_vectors_from_membrane_left_y(q,1)/Protrusion_Histogram_left(q);
Instantaneous_Average_UnitVector_left_x(q,1) =
Protrusion_unit_vectors_left_x(q,1)/Protrusion_Histogram_left(q);
Instantaneous_Average_UnitVector_left_y(q,1) =
Protrusion_unit_vectors_left_y(q,1)/Protrusion_Histogram_left(q);

Instantaneous_Average_Angle_left(q,1) =
atan(Instantaneous_Average_Vector_left_y(q,1)/Instantaneous_Average_Vector_left_x(q,1));
Instantaneous_Average_Unit_Angle_left(q,1) =
atan(Instantaneous_Average_UnitVector_left_y(q,1)/Instantaneous_Average_UnitVector_left_x(q,1));

%Instantaneous_summed_angle(q,1) =
atan(Protrusion_vectors_from_membrane_y(q,1)/Protrusion_vectors_from_membrane_x(q,1));

if Instantaneous_Average_Vector_left_x(q,1) < 0
    Instantaneous_Average_Angle_left(q,1) = Instantaneous_Average_Angle_left(q,1) + pi;
end
if Instantaneous_Average_Vector_left_x(q,1) > 0 && Instantaneous_Average_Vector_left_y(q,1) < 0
    Instantaneous_Average_Angle_left(q,1) = Instantaneous_Average_Angle_left(q,1) + 2*pi;
end
if Instantaneous_Average_Vector_left_x(q,1) == 0 && Instantaneous_Average_Vector_left_y(q,1) < 0
    Instantaneous_Average_Angle_left(q,1) = 3*pi/2;
end
if Instantaneous_Average_Vector_left_x(q,1) == 0 && Instantaneous_Average_Vector_left_y(q,1) > 0
    Instantaneous_Average_Angle_left(q,1) = pi/2;
end

if Instantaneous_Average_UnitVector_left_x(q,1) < 0
    Instantaneous_Average_Unit_Angle_left(q,1) = Instantaneous_Average_Unit_Angle_left(q,1) + pi;
end
if Instantaneous_Average_UnitVector_left_x(q,1) > 0 && Instantaneous_Average_UnitVector_left_y(q,1) <
0
    Instantaneous_Average_Unit_Angle_left(q,1) = Instantaneous_Average_Unit_Angle_left(q,1) + 2*pi;
end
if Instantaneous_Average_UnitVector_left_x(q,1) == 0 && Instantaneous_Average_UnitVector_left_y(q,1)
< 0
    Instantaneous_Average_Unit_Angle_left(q,1) = 3*pi/2;
end
if Instantaneous_Average_UnitVector_left_x(q,1) == 0 && Instantaneous_Average_UnitVector_left_y(q,1)
> 0
    Instantaneous_Average_Unit_Angle_left(q,1) = pi/2;
end
end
end

%[rows_left, cols_left] = find(Protrusion_Histogram_left == 0);
%Instantaneous_Average_Angle_left(rows_left) = [];
%Instantaneous_Average_Unit_Angle_left(rows_left) = [];

```

```

%%%%%% Time Correlation Function for Length Weighted Protrusions %%%
Instantaneous_Average_Angle_left = Instantaneous_Average_Angle_left';

```

```

[lags_length_weighted_left, c_length_weighted_left] =
cxcorr(Instantaneous_Average_Angle_left, Instantaneous_Average_Angle_left);
%lags_length_weighted = lags_length_weighted';
%[rows_length_weighted, columns_length_weighted] = find(lags_length_weighted == 0);
%lags_length_weighted(1:rows_length_weighted - 1) = [];
%c_length_weighted(1:rows_length_weighted - 1) = [];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Time Correlation Function for UnitCircle Protrusions %%%%%%%%%%%%%%
Instantaneous_Average_Unit_Angle_left = Instantaneous_Average_Unit_Angle_left';
[lags_UnitCircle_left, c_UnitCircle_left] =
cxcorr(Instantaneous_Average_Unit_Angle_left, Instantaneous_Average_Unit_Angle_left);
%lags_UnitCircle = lags_UnitCircle';
%[rows_UnitCircle, columns_UnitCircle] = find(lags_UnitCircle == 0);
%lags_UnitCircle(1:rows_UnitCircle - 1) = [];
%c_UnitCircle(1:rows_UnitCircle - 1) = [];
figure()
plot(lags_length_weighted_left, c_length_weighted_left, 'or', lags_UnitCircle_left,
c_UnitCircle_left, 'ob', lags_length_weighted_left, c_length_weighted_left, '-r', lags_UnitCircle_left,
c_UnitCircle_left, '-b')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Fourier Transform of Time Correlation Functions %%%%%%%%%%

fs = (1/60*interval);
NFFT_left = 2^nextpow2(length(Instantaneous_Average_Angle_left));
F_length_weighted_left = fft(c_length_weighted_left, NFFT_left);
x_axis_left = (0:NFFT_left - 1)*(fs/NFFT_left);
F_length_weighted_left = F_length_weighted_left.*conj(F_length_weighted_left)/NFFT_left; %
2*abs(F_length_weighted(1:NFFT/2+1));

F_UnitCircle_left = fft(c_UnitCircle_left, NFFT_left);
F_UnitCircle_left = F_UnitCircle_left.*conj(F_UnitCircle_left)/NFFT_left; % 2*abs(F_UnitCircle(1:NFFT/2+1));

figure()
plot(x_axis_left, F_length_weighted_left, 'or', x_axis_left, F_UnitCircle_left, 'ob', x_axis_left, F_length_weighted_left, '-r',
x_axis_left, F_UnitCircle_left, '-b')
xlabel('Frequency_left (Hz)')
ylabel('Power_left')

Instantaneous_Average_Angle_left = Instantaneous_Average_Angle_left';
Instantaneous_Average_Unit_Angle_left = Instantaneous_Average_Unit_Angle_left';

for q = 1:i
    for m = 1:w_right
        if Protrusions_slice_right(m) == q
            Protrusion_vectors_from_membrane_right_x(q,1) = Protrusion_vectors_from_membrane_right_x(q,1) +
x_right(m,1);
            Protrusion_vectors_from_membrane_right_y(q,1) = Protrusion_vectors_from_membrane_right_y(q,1) +
y_right(m,1);
            Protrusion_unit_vectors_right_x(q,1) = Protrusion_unit_vectors_right_x(q,1) + x_UnitCircle_right(m,1);
            Protrusion_unit_vectors_right_y(q,1) = Protrusion_unit_vectors_right_y(q,1) + y_UnitCircle_right(m,1);
        end
    end
end

```

```

if Protrusion_Histogram_right(q) > 0
    Instantaneous_Average_Vector_right_x(q,1) =
Protrusion_vectors_from_membrane_right_x(q,1)/Protrusion_Histogram_right(q);
    Instantaneous_Average_Vector_right_y(q,1) =
Protrusion_vectors_from_membrane_right_y(q,1)/Protrusion_Histogram_right(q);
    Instantaneous_Average_UnitVector_right_x(q,1) =
Protrusion_unit_vectors_right_x(q,1)/Protrusion_Histogram_right(q);
    Instantaneous_Average_UnitVector_right_y(q,1) =
Protrusion_unit_vectors_right_y(q,1)/Protrusion_Histogram_right(q);

    Instantaneous_Average_Angle_right(q,1) =
atan(Instantaneous_Average_Vector_right_y(q,1)/Instantaneous_Average_Vector_right_x(q,1));
    Instantaneous_Average_Unit_Angle_right(q,1) =
atan(Instantaneous_Average_UnitVector_right_y(q,1)/Instantaneous_Average_UnitVector_right_x(q,1));

    %Instantaneous_summed_angle(q,1) =
atan(Protrusion_vectors_from_membrane_y(q,1)/Protrusion_vectors_from_membrane_x(q,1));

    if Instantaneous_Average_Vector_right_x(q,1) < 0
        Instantaneous_Average_Angle_right(q,1) = Instantaneous_Average_Angle_right(q,1) + pi;
    end
    if Instantaneous_Average_Vector_right_x(q,1) > 0 && Instantaneous_Average_Vector_right_y(q,1) < 0
        Instantaneous_Average_Angle_right(q,1) = Instantaneous_Average_Angle_right(q,1) + 2*pi;
    end
    if Instantaneous_Average_Vector_right_x(q,1) == 0 && Instantaneous_Average_Vector_right_y(q,1) < 0
        Instantaneous_Average_Angle_right(q,1) = 3*pi/2;
    end
    if Instantaneous_Average_Vector_right_x(q,1) == 0 && Instantaneous_Average_Vector_right_y(q,1) > 0
        Instantaneous_Average_Angle_right(q,1) = pi/2;
    end

    if Instantaneous_Average_UnitVector_right_x(q,1) < 0
        Instantaneous_Average_Unit_Angle_right(q,1) = Instantaneous_Average_Unit_Angle_right(q,1) + pi;
    end
    if Instantaneous_Average_UnitVector_right_x(q,1) > 0 &&
Instantaneous_Average_UnitVector_right_y(q,1) < 0
        Instantaneous_Average_Unit_Angle_right(q,1) = Instantaneous_Average_Unit_Angle_right(q,1) + 2*pi;
    end
    if Instantaneous_Average_UnitVector_right_x(q,1) == 0 &&
Instantaneous_Average_UnitVector_right_y(q,1) < 0
        Instantaneous_Average_Unit_Angle_right(q,1) = 3*pi/2;
    end
    if Instantaneous_Average_UnitVector_right_x(q,1) == 0 &&
Instantaneous_Average_UnitVector_right_y(q,1) > 0
        Instantaneous_Average_Unit_Angle_right(q,1) = pi/2;
    end
end
end

%[rows_right, cols_right] = find(Protrusion_Histogram_right == 0);
%Instantaneous_Average_Angle_right(rows_right) = [];
%Instantaneous_Average_Unit_Angle_right(rows_right) = [];

```

%%%%%%%% Time Correlation Function for Length Weighted Protrusions %%%%%%%%%

```

Instantaneous_Average_Angle_right = Instantaneous_Average_Angle_right';
[lags_length_weighted_right, c_length_weighted_right] =
cxcorr(Instantaneous_Average_Angle_right,Instantaneous_Average_Angle_right);
%lags_length_weighted = lags_length_weighted';
%[rows_length_weighted, columns_length_weighted] = find(lags_length_weighted == 0);
%lags_length_weighted(1:rows_length_weighted - 1) = [];
%c_length_weighted(1:rows_length_weighted - 1) = [];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Time Correlation Function for UnitCircle Protrusions %%%%%%%%%%%%%%
Instantaneous_Average_Unit_Angle_right = Instantaneous_Average_Unit_Angle_right';
[lags_UnitCircle_right, c_UnitCircle_right] =
cxcorr(Instantaneous_Average_Unit_Angle_right,Instantaneous_Average_Unit_Angle_right);
%lags_UnitCircle = lags_UnitCircle';
%[rows_UnitCircle, columns_UnitCircle] = find(lags_UnitCircle == 0);
%lags_UnitCircle(1:rows_UnitCircle - 1) = [];
%c_UnitCircle(1:rows_UnitCircle - 1) = [];
figure()
plot(lags_length_weighted_right,c_length_weighted_right,'or',lags_UnitCircle_right,
c_UnitCircle_right,'ob',lags_length_weighted_right, c_length_weighted_right,'-r',lags_UnitCircle_right,
c_UnitCircle_right,'-b')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Fourier Transform of Time Correlation Functions %%%%%%%%%%%%%%

NFFT_right = 2^nextpow2(length(Instantaneous_Average_Angle_right));
F_length_weighted_right = fft(c_length_weighted_right, NFFT_right);
x_axis_right = (0:NFFT_right - 1)*(fs/NFFT_right);
F_length_weighted_right = F_length_weighted_right.*conj(F_length_weighted_right)/NFFT_right; %
2*abs(F_length_weighted(1:NFFT/2+1));

F_UnitCircle_right = fft(c_UnitCircle_right, NFFT_right);
F_UnitCircle_right = F_UnitCircle_right.*conj(F_UnitCircle_right)/NFFT_right;
%2*abs(F_UnitCircle(1:NFFT/2+1));

figure()
plot(x_axis_right,F_length_weighted_right,'or',x_axis_right,F_UnitCircle_right,'ob',x_axis_right,F_length_weighted
_right,'-r',x_axis_right,F_UnitCircle_right,'-b')
xlabel('Frequency_right (Hz)')
ylabel('Power_right')

Instantaneous_Average_Angle_right = Instantaneous_Average_Angle_right';
Instantaneous_Average_Unit_Angle_right = Instantaneous_Average_Unit_Angle_right';
x_axis_right = x_axis_right';
x_axis_left = x_axis_left';
lags_length_weighted_right = lags_length_weighted_right';
lags_length_weighted_left = lags_length_weighted_left';
lags_UnitCircle_right = lags_UnitCircle_right';
lags_UnitCircle_left = lags_UnitCircle_left';
F_length_weighted_right = F_length_weighted_right';
F_length_weighted_left = F_length_weighted_left';
F_UnitCircle_right = F_UnitCircle_right';
F_UnitCircle_left = F_UnitCircle_left';
c_UnitCircle_right = c_UnitCircle_right';
c_UnitCircle_left = c_UnitCircle_left';

```

```

save('x-Fourier_left.txt','x_axis_left','-ascii','-double');
save('x-Fourier_right.txt','x_axis_right','-ascii','-double');
save('Fourier_LW_TimeCorrelation_left.txt','F_length_weighted_left','-ascii','-double');
save('Fourier_UnitCircle_TimeCorrelation_left.txt','F_UnitCircle_left','-ascii','-double');
save('lags_LW_left.txt','lags_length_weighted_left','-ascii','-double');
save('lags_UnitCircle_left.txt','lags_UnitCircle_left','-ascii','-double');
save('Correlation_Sequence_LW_Left.txt','c_length_weighted_left','-ascii','-double');
save('Correlation_Sequence_UnitCircle_left.txt','c_UnitCircle_left','-ascii','-double');
save('Instantaneous_Average_Angle_left.txt','Instantaneous_Average_Angle_left','-ascii','-double');
save('Instantaneous_Average_Unit_Angle_left.txt','Instantaneous_Average_Unit_Angle_left','-ascii','-double');
save('Protrusion_Length_Over_Time_left.txt','instantaneous_average_protrusion_length_left','-ascii','-double');

```

```

save('Fourier_LW_TimeCorrelation_right.txt','F_length_weighted_right','-ascii','-double');
save('Fourier_UnitCircle_TimeCorrelation_right.txt','F_UnitCircle_right','-ascii','-double');
save('lags_LW_right.txt','lags_length_weighted_right','-ascii','-double');
save('lags_UnitCircle_right.txt','lags_UnitCircle_right','-ascii','-double');
save('Correlation_Sequence_LW_right.txt','c_length_weighted_right','-ascii','-double');
save('Correlation_Sequence_UnitCircle_right.txt','c_UnitCircle_right','-ascii','-double');
save('Instantaneous_Average_Angle_right.txt','Instantaneous_Average_Angle_right','-ascii','-double');
save('Instantaneous_Average_Unit_Angle_right.txt','Instantaneous_Average_Unit_Angle_right','-ascii','-double');
save('Protrusion_Length_Over_Time_right.txt','instantaneous_average_protrusion_length_right','-ascii','-double');

```

```

%%%%%%%%%% EARLY TIME ANALYSIS
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%

```

```

rows_right_early = find(Protrusions_slice_right <= 72);
x_P_double_prime_right_early = x_right(rows_right_early);
y_P_double_prime_right_early = y_right(rows_right_early);

rows_left_early = find(Protrusions_slice_left <= 72); %round(length(Uncorrected_Protrusions_x_left)/2));
x_P_double_prime_left_early = x_left(rows_left_early);
y_P_double_prime_left_early = y_left(rows_left_early);

w_left = length(rows_left_early)
F_2_w_left_minus_2 = input('What is the F-distribution value for the left cell?: ');
w_right = length(rows_right_early)
F_2_w_right_minus_2 = input('What is the F-distribution value for the right cell?: ');
t_w_left_minus_1 = input('What is the t-distribution value for the left cell?: ');
t_w_right_minus_1 = input('What is the t-distribution value for the right cell?: ');

```

```

%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%

```

```

%%%%%%%% Calculating angle of protrusion in the appropriate coordinate
%%%%%%%% system and correcting for quadrant location

```

```

Protrusion_angle_left = atan(y_P_double_prime_left_early./x_P_double_prime_left_early);
Protrusions_Unit_Circle_x_left = cos(Protrusion_angle_left);
Protrusions_Unit_Circle_y_left = sin(Protrusion_angle_left);

```

```

for m = 1:w_left

```

```

    if x_P_double_prime_left_early(m,1) < 0
        Protrusion_angle_left(m,1) = Protrusion_angle_left(m,1) + pi;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
    if x_P_double_prime_left_early(m,1) > 0 && y_P_double_prime_left_early(m,1) < 0
        Protrusion_angle_left(m,1) = Protrusion_angle_left(m,1) + 2*pi;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
    if x_P_double_prime_left_early(m,1) == 0 && y_P_double_prime_left_early(m,1) < 0
        Protrusion_angle_left(m,1) = 3*pi/2;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
    if x_P_double_prime_left_early(m,1) == 0 && y_P_double_prime_left_early(m,1) > 0
        Protrusion_angle_left(m,1) = pi/2;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
end
circ_plot(Protrusion_angle_left);
title('Left Cell Early');
save('Protrusion_angles_early_left.txt','Protrusion_angle_left','-ascii','-double');

```

```

Protrusion_angle_left
%%%%%%%%

```

```

Protrusion_angle_right = atan(y_P_double_prime_right_early./x_P_double_prime_right_early);
Protrusions_Unit_Circle_x_right = cos(Protrusion_angle_right);
Protrusions_Unit_Circle_y_right = sin(Protrusion_angle_right);

```

```

for m = 1:w_right

```

```

    if x_P_double_prime_right_early(m,1) < 0
        Protrusion_angle_right(m,1) = Protrusion_angle_right(m,1) + pi;
        Protrusions_Unit_Circle_x_right(m,1) = cos(Protrusion_angle_right(m,1));
        Protrusions_Unit_Circle_y_right(m,1) = sin(Protrusion_angle_right(m,1));
    end
    if x_P_double_prime_right_early(m,1) > 0 && y_P_double_prime_right_early(m,1) < 0
        Protrusion_angle_right(m,1) = Protrusion_angle_right(m,1) + 2*pi;
        Protrusions_Unit_Circle_x_right(m,1) = cos(Protrusion_angle_right(m,1));
        Protrusions_Unit_Circle_y_right(m,1) = sin(Protrusion_angle_right(m,1));
    end
    if x_P_double_prime_right_early(m,1) == 0 && y_P_double_prime_right_early(m,1) < 0
        Protrusion_angle_right(m,1) = 3*pi/2;

```

```

    Protrusions_Unit_Circle_x_right(m,1) = cos(Protrusion_angle_right(m,1));
    Protrusions_Unit_Circle_y_right(m,1) = sin(Protrusion_angle_right(m,1));
end
if x_P_double_prime_right_early(m,1) == 0 && y_P_double_prime_right_early(m,1) > 0
    Protrusion_angle_right(m,1) = pi/2;
    Protrusions_Unit_Circle_x_right(m,1) = cos(Protrusion_angle_right(m,1));
    Protrusions_Unit_Circle_y_right(m,1) = sin(Protrusion_angle_right(m,1));
end
end
figure()
circ_plot(Protrusion_angle_right);
title('Right Cell Early');
save('Protrusion_angles_early_right.txt','Protrusion_angle_right','-ascii','-double');

Binned_protrusions_30_left_early = zeros(12,1);
Binned_protrusions_180_left_early = zeros(2,1);
Binned_protrusions_30_right_early = zeros(12,1);
Binned_protrusions_180_right_early = zeros(2,1);

for m = 1:w_left
    if 0 <= Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= pi/6
        Binned_protrusions_30_left_early(1) = Binned_protrusions_30_left_early(1) + 1;
        Binned_protrusions_180_left_early(1) = Binned_protrusions_180_left_early(1) + 1;
    end

    if pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= pi/3
        Binned_protrusions_30_left_early(2) = Binned_protrusions_30_left_early(2) + 1;
        Binned_protrusions_180_left_early(1) = Binned_protrusions_180_left_early(1) + 1;
    end

    if pi/3 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= pi/2
        Binned_protrusions_30_left_early(3) = Binned_protrusions_30_left_early(3) + 1;
        Binned_protrusions_180_left_early(1) = Binned_protrusions_180_left_early(1) + 1;
    end

    if pi/2 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 2*pi/3
        Binned_protrusions_30_left_early(4) = Binned_protrusions_30_left_early(4) + 1;
        Binned_protrusions_180_left_early(2) = Binned_protrusions_180_left_early(2) + 1;
    end

    if 2*pi/3 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 5*pi/6
        Binned_protrusions_30_left_early(5) = Binned_protrusions_30_left_early(5) + 1;
        Binned_protrusions_180_left_early(2) = Binned_protrusions_180_left_early(2) + 1;
    end

    if 5*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= pi
        Binned_protrusions_30_left_early(6) = Binned_protrusions_30_left_early(6) + 1;
        Binned_protrusions_180_left_early(2) = Binned_protrusions_180_left_early(2) + 1;
    end

    if pi < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 7*pi/6
        Binned_protrusions_30_left_early(7) = Binned_protrusions_30_left_early(7) + 1;
        Binned_protrusions_180_left_early(2) = Binned_protrusions_180_left_early(2) + 1;
    end
end

```

```

if 7*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 8*pi/6
    Binned_protrusions_30_left_early(8) = Binned_protrusions_30_left_early(8) + 1;
    Binned_protrusions_180_left_early(2) = Binned_protrusions_180_left_early(2) + 1;
end

if 8*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 9*pi/6
    Binned_protrusions_30_left_early(9) = Binned_protrusions_30_left_early(9) + 1;
    Binned_protrusions_180_left_early(2) = Binned_protrusions_180_left_early(2) + 1;
end

if 9*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 10*pi/6
    Binned_protrusions_30_left_early(10) = Binned_protrusions_30_left_early(10) + 1;
    Binned_protrusions_180_left_early(1) = Binned_protrusions_180_left_early(1) + 1;
end

if 10*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 11*pi/6
    Binned_protrusions_30_left_early(11) = Binned_protrusions_30_left_early(11) + 1;
    Binned_protrusions_180_left_early(1) = Binned_protrusions_180_left_early(1) + 1;
end

if 11*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 2*pi
    Binned_protrusions_30_left_early(12) = Binned_protrusions_30_left_early(12) + 1;
    Binned_protrusions_180_left_early(1) = Binned_protrusions_180_left_early(1) + 1;
end
end

for m = 1:w_right
    if 0 <= Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= pi/6
        Binned_protrusions_30_right_early(1) = Binned_protrusions_30_right_early(1) + 1;
        Binned_protrusions_180_right_early(1) = Binned_protrusions_180_right_early(1) + 1;
    end

    if pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= pi/3
        Binned_protrusions_30_right_early(2) = Binned_protrusions_30_right_early(2) + 1;
        Binned_protrusions_180_right_early(1) = Binned_protrusions_180_right_early(1) + 1;
    end

    if pi/3 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= pi/2
        Binned_protrusions_30_right_early(3) = Binned_protrusions_30_right_early(3) + 1;
        Binned_protrusions_180_right_early(1) = Binned_protrusions_180_right_early(1) + 1;
    end

    if pi/2 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 2*pi/3
        Binned_protrusions_30_right_early(4) = Binned_protrusions_30_right_early(4) + 1;
        Binned_protrusions_180_right_early(2) = Binned_protrusions_180_right_early(2) + 1;
    end

    if 2*pi/3 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 5*pi/6
        Binned_protrusions_30_right_early(5) = Binned_protrusions_30_right_early(5) + 1;
        Binned_protrusions_180_right_early(2) = Binned_protrusions_180_right_early(2) + 1;
    end

    if 5*pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= pi
        Binned_protrusions_30_right_early(6) = Binned_protrusions_30_right_early(6) + 1;
        Binned_protrusions_180_right_early(2) = Binned_protrusions_180_right_early(2) + 1;
    end
end

```



```

end

if pi < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 7*pi/6
    Binned_protrusions_30_right_early(7) = Binned_protrusions_30_right_early(7) + 1;
    Binned_protrusions_180_right_early(2) = Binned_protrusions_180_right_early(2) + 1;
end

if 7*pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 8*pi/6
    Binned_protrusions_30_right_early(8) = Binned_protrusions_30_right_early(8) + 1;
    Binned_protrusions_180_right_early(2) = Binned_protrusions_180_right_early(2) + 1;
end

if 8*pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 9*pi/6
    Binned_protrusions_30_right_early(9) = Binned_protrusions_30_right_early(9) + 1;
    Binned_protrusions_180_right_early(2) = Binned_protrusions_180_right_early(2) + 1;
end

if 9*pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 10*pi/6
    Binned_protrusions_30_right_early(10) = Binned_protrusions_30_right_early(10) + 1;
    Binned_protrusions_180_right_early(1) = Binned_protrusions_180_right_early(1) + 1;
end

if 10*pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 11*pi/6
    Binned_protrusions_30_right_early(11) = Binned_protrusions_30_right_early(11) + 1;
    Binned_protrusions_180_right_early(1) = Binned_protrusions_180_right_early(1) + 1;
end

if 11*pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= 2*pi
    Binned_protrusions_30_right_early(12) = Binned_protrusions_30_right_early(12) + 1;
    Binned_protrusions_180_right_early(1) = Binned_protrusions_180_right_early(1) + 1;
end
end

%%%%%%

x_UnitCircle_left = Protrusions_Unit_Circle_x_left;
y_UnitCircle_left = Protrusions_Unit_Circle_y_left;
x_UnitCircle_right = Protrusions_Unit_Circle_x_right;
y_UnitCircle_right = Protrusions_Unit_Circle_y_right;

y_UnitCircle_avg_left = mean(y_UnitCircle_left);
x_UnitCircle_avg_left = mean(x_UnitCircle_left);
y_UnitCircle_avg_right = mean(y_UnitCircle_right);
x_UnitCircle_avg_right = mean(x_UnitCircle_right);

resultant_length_UnitCircle_left = sqrt(x_UnitCircle_avg_left^2 + y_UnitCircle_avg_left^2);
resultant_length_UnitCircle_right = sqrt(x_UnitCircle_avg_right^2 + y_UnitCircle_avg_right^2);

std_UnitCircle_x_left = std(x_UnitCircle_left);
std_UnitCircle_y_left = std(y_UnitCircle_left);
std_UnitCircle_x_right = std(x_UnitCircle_right);
std_UnitCircle_y_right = std(y_UnitCircle_right);

Theta_Ave_left = atan(y_UnitCircle_avg_left/x_UnitCircle_avg_left);
if x_UnitCircle_avg_left < 0

```

```

    Theta_Ave_left = Theta_Ave_left + pi;
end
if x_UnitCircle_avg_left > 0 && y_UnitCircle_avg_left < 0
    Theta_Ave_left = Theta_Ave_left + 2*pi;
end
if x_UnitCircle_avg_left == 0 && y_UnitCircle_avg_left < 0
    Theta_Ave_left = 3*pi/2;
end
if x_UnitCircle_avg_left == 0 && y_UnitCircle_avg_left > 0
    Theta_Ave_left = pi/2;
end

Theta_Ave_right = atan(y_UnitCircle_avg_right/x_UnitCircle_avg_right);
if x_UnitCircle_avg_right < 0
    Theta_Ave_right = Theta_Ave_right + pi;
end
if x_UnitCircle_avg_right > 0 && y_UnitCircle_avg_right < 0
    Theta_Ave_right = Theta_Ave_right + 2*pi;
end
if x_UnitCircle_avg_right == 0 && y_UnitCircle_avg_right < 0
    Theta_Ave_right = 3*pi/2;
end
if x_UnitCircle_avg_right == 0 && y_UnitCircle_avg_right > 0
    Theta_Ave_right = pi/2;
end

%%%%%%%%%%%%%% V-Test
%%%%%%%%%%%%%%
for phi = (1:360)
    V_left(phi) = w_left*resultant_length_UnitCircle_left*cos(Theta_Ave_left - phi*pi/180);
    V_right(phi) = w_right*resultant_length_UnitCircle_right*cos(Theta_Ave_right - phi*pi/180);
end
u_left = V_left*sqrt(2/w_left);
u_right = V_right*sqrt(2/w_right);
u_left = u_left';
u_right = u_right';

%%%%%%%%%%%%%% Rao's Spacing Test %%%%%%%%%%%%%%%
Theta_Ave_left = Theta_Ave_left*180/pi;
Protrusion_angle_left
Protrusion_angle_left = Protrusion_angle_left*180/pi;
Protrusion_angle_left
w_left
T_left = []
lambda_left = 360/w_left    %degrees hereon
sorted_angles_left = sort(Protrusion_angle_left);
T_left(1) = sorted_angles_left(2) - sorted_angles_left(1);
for i = 2:(w_left - 1)
    T_left(i) = sorted_angles_left(i + 1) - sorted_angles_left(i);
end
T_left = T_left';
T_left
U_argument_left = abs(T_left - lambda_left);
U_left = 1/2*sum(U_argument_left);
Protrusion_angle_left

```

```

Theta_Ave_right = Theta_Ave_right*180/pi;
Protrusion_angle_right = Protrusion_angle_right*180/pi;

T_right = []
lambda_right = 360/w_right %degrees hereon
sorted_angles_right = sort(Protrusion_angle_right);
T_right(1) = sorted_angles_right(2) - sorted_angles_right(1);
for i = 2:(w_right - 1)
    T_right(i) = sorted_angles_right(i + 1) - sorted_angles_right(i);
end
T_right = T_right';
U_argument_right = abs(T_right - lambda_right);
U_right = 1/2*sum(U_argument_right);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Rayleigh's Test %%%%%%%%%
z_left = w_left*resultant_length_UnitCircle_left^2;
z_right = w_right*resultant_length_UnitCircle_right^2;

P_left = exp(sqrt(1 + 4*w_left + 4*(w_left^2 - (w_left*resultant_length_UnitCircle_left)^2)) - (1 + 2*w_left));
P_right = exp(sqrt(1 + 4*w_right + 4*(w_right^2 - (w_right*resultant_length_UnitCircle_right)^2)) - (1 + 2*w_right));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Confidence Interval for Average Angles on Unit Circle %%%%%%%%%
chi_square = 3.841; %95 percent confidence_interval
if resultant_length_UnitCircle_left <= 0.9 && resultant_length_UnitCircle_left > sqrt(chi_square/(2*w_left))
    d_unit_circle_left = ceil((180/pi)*acos(sqrt((2*w_left*(2*(w_left*resultant_length_UnitCircle_left)^2 - w_left*chi_square))/(4*w_left - chi_square)))/w_left*resultant_length_UnitCircle_left));
    save('Confidence_Interval_UnitCircle_left EARLY.txt','d_unit_circle_left','-ascii','-double');
end
if resultant_length_UnitCircle_right <= 0.9 && resultant_length_UnitCircle_right > sqrt(chi_square/(2*w_right))
    d_unit_circle_right = ceil((180/pi)*acos(sqrt((2*w_right*(2*(w_right*resultant_length_UnitCircle_right)^2 - w_right*chi_square))/(4*w_right - chi_square)))/w_right*resultant_length_UnitCircle_right));
    save('Confidence_Interval_UnitCircle_right EARLY.txt','d_unit_circle_right','-ascii','-double');
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

x_confidence_interval_UnitCircle_left = t_w_left_minus_1*std_UnitCircle_x_left/sqrt(w_left);
y_confidence_interval_UnitCircle_left = t_w_left_minus_1*std_UnitCircle_y_left/sqrt(w_left);
x_confidence_interval_UnitCircle_right = t_w_right_minus_1*std_UnitCircle_x_right/sqrt(w_right);
y_confidence_interval_UnitCircle_right = t_w_right_minus_1*std_UnitCircle_y_right/sqrt(w_right);

summary_UnitCircle_left = [initial_separation Theta_Ave_left resultant_length_UnitCircle_left w_left P_left U_left
x_UnitCircle_avg_left std_UnitCircle_x_left x_confidence_interval_UnitCircle_left y_UnitCircle_avg_left
std_UnitCircle_y_left y_confidence_interval_UnitCircle_left];
summary_UnitCircle_right = [initial_separation Theta_Ave_right resultant_length_UnitCircle_right w_right P_right
U_right x_UnitCircle_avg_right std_UnitCircle_x_right x_confidence_interval_UnitCircle_right
y_UnitCircle_avg_right std_UnitCircle_y_right y_confidence_interval_UnitCircle_right];

save('Summary UnitCircle Left Early.txt','summary_UnitCircle_left','-ascii','-double');
save('Summary UnitCircle Right Early.txt','summary_UnitCircle_right','-ascii','-double');
save('Critical_Value_for_V-Test(u)_Left Early.txt','u_left','-ascii','-double');
save('Critical_Value_for_V-Test(u)_Right Early.txt','u_right','-ascii','-double');
%save('x-coordinates_protrusions_left.txt','x_UnitCircle_left','-ascii','-double');
%save('y-coordinates_protrusions_left.txt','y_UnitCircle_left','-ascii','-double');

```

```

%save('x-coordinates_protrusions_right.txt','x_UnitCircle_right','-ascii','-double');
%save('y-coordinates_protrusions_right.txt','y_UnitCircle_right','-ascii','-double');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

x_left = x_P_double_prime_left_early;
y_left = y_P_double_prime_left_early;
x_right = x_P_double_prime_right_early;
y_right = y_P_double_prime_right_early;

y_avg_left = mean(y_left);
x_avg_left = mean(x_left);
y_avg_right = mean(y_right);
x_avg_right = mean(x_right);

std_x_left = std(x_left);
std_y_left = std(y_left);
std_x_right = std(x_right);
std_y_right = std(y_right);

s1_squared_left = sum((x_left - x_avg_left).^2*(1/(w_left - 1))); % [length^2]
s2_squared_left = sum((y_left - y_avg_left).^2*(1/(w_left - 1))); % [length^2]
s1_squared_right = sum((x_right - x_avg_right).^2*(1/(w_right - 1))); % [length^2]
s2_squared_right = sum((y_right - y_avg_right).^2*(1/(w_right - 1))); % [length^2]

r_left = sum((x_left - x_avg_left).*(y_left - y_avg_left))/((w_left - 1)*sqrt(s1_squared_left)*sqrt(s2_squared_left));
% [dimensionless]
T_squared_confidence_left = 2*((w_left - 1)/(w_left - 2))*F_2_w_left_minus_2;
T_squared_left = (w_left/(1 - r_left^2))*(x_avg_left^2/s1_squared_left -
2*r_left*x_avg_left*y_avg_left/(sqrt(s1_squared_left)*sqrt(s2_squared_left)) + y_avg_left^2/s2_squared_left);
r_right = sum((x_right - x_avg_right).*(y_right - y_avg_right))/((w_right -
1)*sqrt(s1_squared_right)*sqrt(s2_squared_right)); % [dimensionless]
T_squared_confidence_right = 2*((w_right - 1)/(w_right - 2))*F_2_w_right_minus_2;
T_squared_right = (w_right/(1 - r_right^2))*(x_avg_right^2/s1_squared_right -
2*r_right*x_avg_right*y_avg_right/(sqrt(s1_squared_right)*sqrt(s2_squared_right)) +
y_avg_right^2/s2_squared_right);

A_left = s2_squared_left; % [length^2]
A_right = s2_squared_right; % [length^2]
B_left = -r_left*sqrt(s1_squared_left)*sqrt(s2_squared_left); % [length^2]
B_right = -r_right*sqrt(s1_squared_right)*sqrt(s2_squared_right); % [length^2]
C_left = s1_squared_left; % [length^2]
C_right = s1_squared_right; % [length^2]
D_left = (1 - r_left^2)*s1_squared_left*s2_squared_left; % [length^4]
D_right = (1 - r_right^2)*s1_squared_right*s2_squared_right; % [length^4]
R_left = sqrt((A_left - C_left)^2 + 4*B_left^2); % [length^2]
R_right = sqrt((A_right - C_right)^2 + 4*B_right^2); % [length^2]
a_left = sqrt(2*D_left/(A_left + C_left - R_left)); % [length]
a_right = sqrt(2*D_right/(A_right + C_right - R_right)); % [length]
b_left = sqrt(2*D_left/(A_left + C_left + R_left)); % [length]
b_right = sqrt(2*D_right/(A_right + C_right + R_right)); % [length]
theta_left = atan(2*B_left/(A_left - C_left - R_left));

```

```

theta_right = atan(2*B_right/(A_right - C_right - R_right));

H_left = A_left*C_left - B_left^2;
H_right = A_right*C_right - B_right^2;
G_left = A_left*x_avg_left^2 + 2*B_left*x_avg_left*y_avg_left + C_left*y_avg_left^2 - D_left;
G_right = A_right*x_avg_right^2 + 2*B_right*x_avg_right*y_avg_right + C_right*y_avg_right^2 - D_right;
U_left = (H_left*x_avg_left^2 - C_left*D_left)^(-1);
U_right = (H_right*x_avg_right^2 - C_right*D_right)^(-1);
V_left = (D_left*G_left*H_left)^(1/2);
V_right = (D_right*G_right*H_right)^(1/2);
W_left = H_left*x_avg_left*y_avg_left + B_left*D_left;
W_right = H_right*x_avg_right*y_avg_right + B_right*D_right;
m1_left = U_left*(W_left + V_left);
m2_left = U_left*(W_left - V_left);
m1_right = U_right*(W_right + V_right);
m2_right = U_right*(W_right - V_right);
m1_theta_left = (180/pi)*atan(m1_left);
m2_theta_left = (180/pi)*atan(m2_left);
m1_theta_right = (180/pi)*atan(m1_right);
m2_theta_right = (180/pi)*atan(m2_right);

D_confidence_left = (1 - r_left^2)*s1_squared_left*s2_squared_left*(1/w_left)*T_squared_confidence_left;
% [length^4]
a_confidence_left = a_left*(w_left^(-1/2))*sqrt(T_squared_confidence_left);
b_confidence_left = b_left*(w_left^(-1/2))*sqrt(T_squared_confidence_left);
D_confidence_right = (1 -
r_right^2)*s1_squared_right*s2_squared_right*(1/w_right)*T_squared_confidence_right; % [length^4]
a_confidence_right = a_right*(w_right^(-1/2))*sqrt(T_squared_confidence_right);
b_confidence_right = b_right*(w_right^(-1/2))*sqrt(T_squared_confidence_right);

phi = (1:360)';
phi = phi*(pi/180);
X_left = x_avg_left + a_left*cos(theta_left)*cos(phi) - b_left*sin(theta_left)*sin(phi);
Y_left = y_avg_left + a_left*sin(theta_left)*cos(phi) + b_left*cos(theta_left)*sin(phi);
X_confidence_left = x_avg_left + a_confidence_left*cos(theta_left)*cos(phi) -
b_confidence_left*sin(theta_left)*sin(phi);
Y_confidence_left = y_avg_left + a_confidence_left*sin(theta_left)*cos(phi) +
b_confidence_left*cos(theta_left)*sin(phi);
X_right = x_avg_right + a_right*cos(theta_right)*cos(phi) - b_right*sin(theta_right)*sin(phi);
Y_right = y_avg_right + a_right*sin(theta_right)*cos(phi) + b_right*cos(theta_right)*sin(phi);
X_confidence_right = x_avg_right + a_confidence_right*cos(theta_right)*cos(phi) -
b_confidence_right*sin(theta_right)*sin(phi);
Y_confidence_right = y_avg_right + a_confidence_right*sin(theta_right)*cos(phi) +
b_confidence_right*cos(theta_right)*sin(phi);

figure()
plot(X_left,Y_left,'b')
hold on
plot(x_left,y_left,'ro')
plot(X_confidence_left,Y_confidence_left,'k')
hold off

figure()
plot(X_right,Y_right,'b')
hold on

```

```

plot(x_right,y_right,'ro')
plot(X_confidence_right,Y_confidence_right,'.k')
hold off

x_confidence_interval_left = t_w_left_minus_1*std_x_left/sqrt(w_left);
y_confidence_interval_left = t_w_left_minus_1*std_y_left/sqrt(w_left);
x_confidence_interval_right = t_w_right_minus_1*std_x_right/sqrt(w_right);
y_confidence_interval_right = t_w_right_minus_1*std_y_right/sqrt(w_right);

theta_left = theta_left*180/pi;
theta_right = theta_right*180/pi;

avg_resultant_vector_angle_left = atan(y_avg_left/x_avg_left);
avg_resultant_vector_angle_right = atan(y_avg_right/x_avg_right);

if x_avg_left < 0
    avg_resultant_vector_angle_left = avg_resultant_vector_angle_left + pi;
end
if x_avg_left > 0 && y_avg_left < 0
    avg_resultant_vector_angle_left = avg_resultant_vector_angle_left + 2*pi;
end
if x_avg_left == 0 && y_avg_left < 0
    avg_resultant_vector_angle_left = 3*pi/2;
end
if x_avg_left == 0 && y_avg_left > 0
    avg_resultant_vector_angle_left = pi/2;
end

if x_avg_right < 0
    avg_resultant_vector_angle_right = avg_resultant_vector_angle_right + pi;
end
if x_avg_right > 0 && y_avg_right < 0
    avg_resultant_vector_angle_right = avg_resultant_vector_angle_right + 2*pi;
end
if x_avg_right == 0 && y_avg_right < 0
    avg_resultant_vector_angle_right = 3*pi/2;
end
if x_avg_right == 0 && y_avg_right > 0
    avg_resultant_vector_angle_right = pi/2;
end

avg_resultant_vector_angle_left = (180/pi)*avg_resultant_vector_angle_left;
avg_resultant_vector_angle_right = (180/pi)*avg_resultant_vector_angle_right;
mean_vector_magnitude_left = mean(sqrt(x_left.^2 + y_left.^2));
mean_vector_magnitude_right = mean(sqrt(x_right.^2 + y_right.^2));

protrusion_frequency_left = w_left/12;
protrusion_frequency_right = w_right/12;

summary_left = [initial_separation theta_left avg_resultant_vector_angle_left T_squared_left
T_squared_confidence_left a_left w_left m1_theta_left m2_theta_left mean_vector_magnitude_left x_avg_left
std_x_left x_confidence_interval_left y_avg_left std_y_left y_confidence_interval_left protrusion_frequency_left];
summary_right = [initial_separation theta_right avg_resultant_vector_angle_right T_squared_right
T_squared_confidence_right a_right w_right m1_theta_right m2_theta_right mean_vector_magnitude_right

```



```

rows_left_late = find(Protrusions_slice_left > 72); %round(length(Uncorrected_Protrusions_x_left)/2));
x_P_double_prime_left_late = x_left(rows_left_late);
y_P_double_prime_left_late = y_left(rows_left_late);

w_left = length(rows_left_late)
F_2_w_left_minus_2 = input('What is the F-distribution value for the left cell?: ');
w_right = length(rows_right_late)
F_2_w_right_minus_2 = input('What is the F-distribution value for the right cell?: ');
t_w_left_minus_1 = input('What is the t-distribution value for the left cell?: ');
t_w_right_minus_1 = input('What is the t-distribution value for the right cell?: ');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Calculating angle of protrusion in the appropriate coordinate
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% system and correcting for quadrant location

Protrusion_angle_left = atan(y_P_double_prime_left_late./x_P_double_prime_left_late);
Protrusions_Unit_Circle_x_left = cos(Protrusion_angle_left);
Protrusions_Unit_Circle_y_left = sin(Protrusion_angle_left);

for m = 1:w_left

    if x_P_double_prime_left_late(m,1) < 0
        Protrusion_angle_left(m,1) = Protrusion_angle_left(m,1) + pi;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
    if x_P_double_prime_left_late(m,1) > 0 && y_P_double_prime_left_late(m,1) < 0
        Protrusion_angle_left(m,1) = Protrusion_angle_left(m,1) + 2*pi;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
    if x_P_double_prime_left_late(m,1) == 0 && y_P_double_prime_left_late(m,1) < 0
        Protrusion_angle_left(m,1) = 3*pi/2;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
    if x_P_double_prime_left_late(m,1) == 0 && y_P_double_prime_left_late(m,1) > 0
        Protrusion_angle_left(m,1) = pi/2;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
end
end
circ_plot(Protrusion_angle_left);
title('Left Cell Late');
save('Protrusion_angles_late_left.txt','Protrusion_angle_left','-ascii','-double');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Protrusion_angle_right = atan(y_P_double_prime_right_late./x_P_double_prime_right_late);
Protrusions_Unit_Circle_x_right = cos(Protrusion_angle_right);
Protrusions_Unit_Circle_y_right = sin(Protrusion_angle_right);

```



```

for m = 1:w_right

    if x_P_double_prime_right_late(m,1) < 0
        Protrusion_angle_right(m,1) = Protrusion_angle_right(m,1) + pi;
        Protrusions_Unit_Circle_x_right(m,1) = cos(Protrusion_angle_right(m,1));
        Protrusions_Unit_Circle_y_right(m,1) = sin(Protrusion_angle_right(m,1));
    end
    if x_P_double_prime_right_late(m,1) > 0 && y_P_double_prime_right_late(m,1) < 0
        Protrusion_angle_right(m,1) = Protrusion_angle_right(m,1) + 2*pi;
        Protrusions_Unit_Circle_x_right(m,1) = cos(Protrusion_angle_right(m,1));
        Protrusions_Unit_Circle_y_right(m,1) = sin(Protrusion_angle_right(m,1));
    end
    if x_P_double_prime_right_late(m,1) == 0 && y_P_double_prime_right_late(m,1) < 0
        Protrusion_angle_right(m,1) = 3*pi/2;
        Protrusions_Unit_Circle_x_right(m,1) = cos(Protrusion_angle_right(m,1));
        Protrusions_Unit_Circle_y_right(m,1) = sin(Protrusion_angle_right(m,1));
    end
    if x_P_double_prime_right_late(m,1) == 0 && y_P_double_prime_right_late(m,1) > 0
        Protrusion_angle_right(m,1) = pi/2;
        Protrusions_Unit_Circle_x_right(m,1) = cos(Protrusion_angle_right(m,1));
        Protrusions_Unit_Circle_y_right(m,1) = sin(Protrusion_angle_right(m,1));
    end
end
figure()
circ_plot(Protrusion_angle_right);
title('Right Cell Late');
save('Protrusion_angles_late_right.txt','Protrusion_angle_right','-ascii','-double');

Binned_protrusions_30_left_late = zeros(12,1);
Binned_protrusions_180_left_late = zeros(2,1);
Binned_protrusions_30_right_late = zeros(12,1);
Binned_protrusions_180_right_late = zeros(2,1);

for m = 1:w_left
    if 0 <= Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= pi/6
        Binned_protrusions_30_left_late(1) = Binned_protrusions_30_left_late(1) + 1;
        Binned_protrusions_180_left_late(1) = Binned_protrusions_180_left_late(1) + 1;
    end

    if pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= pi/3
        Binned_protrusions_30_left_late(2) = Binned_protrusions_30_left_late(2) + 1;
        Binned_protrusions_180_left_late(1) = Binned_protrusions_180_left_late(1) + 1;
    end

    if pi/3 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= pi/2
        Binned_protrusions_30_left_late(3) = Binned_protrusions_30_left_late(3) + 1;
        Binned_protrusions_180_left_late(1) = Binned_protrusions_180_left_late(1) + 1;
    end

    if pi/2 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 2*pi/3
        Binned_protrusions_30_left_late(4) = Binned_protrusions_30_left_late(4) + 1;
        Binned_protrusions_180_left_late(2) = Binned_protrusions_180_left_late(2) + 1;
    end

    if 2*pi/3 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 5*pi/6

```

```

    Binned_protrusions_30_left_late(5) = Binned_protrusions_30_left_late(5) + 1;
    Binned_protrusions_180_left_late(2) = Binned_protrusions_180_left_late(2) + 1;
end

if 5*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= pi
    Binned_protrusions_30_left_late(6) = Binned_protrusions_30_left_late(6) + 1;
    Binned_protrusions_180_left_late(2) = Binned_protrusions_180_left_late(2) + 1;
end

if pi < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 7*pi/6
    Binned_protrusions_30_left_late(7) = Binned_protrusions_30_left_late(7) + 1;
    Binned_protrusions_180_left_late(2) = Binned_protrusions_180_left_late(2) + 1;
end

if 7*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 8*pi/6
    Binned_protrusions_30_left_late(8) = Binned_protrusions_30_left_late(8) + 1;
    Binned_protrusions_180_left_late(2) = Binned_protrusions_180_left_late(2) + 1;
end

if 8*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 9*pi/6
    Binned_protrusions_30_left_late(9) = Binned_protrusions_30_left_late(9) + 1;
    Binned_protrusions_180_left_late(2) = Binned_protrusions_180_left_late(2) + 1;
end

if 9*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 10*pi/6
    Binned_protrusions_30_left_late(10) = Binned_protrusions_30_left_late(10) + 1;
    Binned_protrusions_180_left_late(1) = Binned_protrusions_180_left_late(1) + 1;
end

if 10*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 11*pi/6
    Binned_protrusions_30_left_late(11) = Binned_protrusions_30_left_late(11) + 1;
    Binned_protrusions_180_left_late(1) = Binned_protrusions_180_left_late(1) + 1;
end

if 11*pi/6 < Protrusion_angle_left(m,1) && Protrusion_angle_left(m,1) <= 2*pi
    Binned_protrusions_30_left_late(12) = Binned_protrusions_30_left_late(12) + 1;
    Binned_protrusions_180_left_late(1) = Binned_protrusions_180_left_late(1) + 1;
end
end

for m = 1:w_right
    if 0 <= Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= pi/6
        Binned_protrusions_30_right_late(1) = Binned_protrusions_30_right_late(1) + 1;
        Binned_protrusions_180_right_late(1) = Binned_protrusions_180_right_late(1) + 1;
    end

    if pi/6 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= pi/3
        Binned_protrusions_30_right_late(2) = Binned_protrusions_30_right_late(2) + 1;
        Binned_protrusions_180_right_late(1) = Binned_protrusions_180_right_late(1) + 1;
    end

    if pi/3 < Protrusion_angle_right(m,1) && Protrusion_angle_right(m,1) <= pi/2
        Binned_protrusions_30_right_late(3) = Binned_protrusions_30_right_late(3) + 1;
        Binned_protrusions_180_right_late(1) = Binned_protrusions_180_right_late(1) + 1;
    end
end

```



```

y_UnitCircle_avg_left = mean(y_UnitCircle_left);
x_UnitCircle_avg_left = mean(x_UnitCircle_left);
y_UnitCircle_avg_right = mean(y_UnitCircle_right);
x_UnitCircle_avg_right = mean(x_UnitCircle_right);

resultant_length_UnitCircle_left = sqrt(x_UnitCircle_avg_left^2 + y_UnitCircle_avg_left^2);
resultant_length_UnitCircle_right = sqrt(x_UnitCircle_avg_right^2 + y_UnitCircle_avg_right^2);

std_UnitCircle_x_left = std(x_UnitCircle_left);
std_UnitCircle_y_left = std(y_UnitCircle_left);
std_UnitCircle_x_right = std(x_UnitCircle_right);
std_UnitCircle_y_right = std(y_UnitCircle_right);

Theta_Ave_left = atan(y_UnitCircle_avg_left/x_UnitCircle_avg_left);
if x_UnitCircle_avg_left < 0
    Theta_Ave_left = Theta_Ave_left + pi;
end
if x_UnitCircle_avg_left > 0 && y_UnitCircle_avg_left < 0
    Theta_Ave_left = Theta_Ave_left + 2*pi;
end
if x_UnitCircle_avg_left == 0 && y_UnitCircle_avg_left < 0
    Theta_Ave_left = 3*pi/2;
end
if x_UnitCircle_avg_left == 0 && y_UnitCircle_avg_left > 0
    Theta_Ave_left = pi/2;
end

Theta_Ave_right = atan(y_UnitCircle_avg_right/x_UnitCircle_avg_right);
if x_UnitCircle_avg_right < 0
    Theta_Ave_right = Theta_Ave_right + pi;
end
if x_UnitCircle_avg_right > 0 && y_UnitCircle_avg_right < 0
    Theta_Ave_right = Theta_Ave_right + 2*pi;
end
if x_UnitCircle_avg_right == 0 && y_UnitCircle_avg_right < 0
    Theta_Ave_right = 3*pi/2;
end
if x_UnitCircle_avg_right == 0 && y_UnitCircle_avg_right > 0
    Theta_Ave_right = pi/2;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% V-Test
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for phi = (1:360)
    V_left(phi) = w_left*resultant_length_UnitCircle_left*cos(Theta_Ave_left - phi*pi/180);
    V_right(phi) = w_right*resultant_length_UnitCircle_right*cos(Theta_Ave_right - phi*pi/180);
end
u_left = V_left*sqrt(2/w_left);
u_right = V_right*sqrt(2/w_right);
u_left = u_left';
u_right = u_right';
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Rao's Spacing Test %%%%%%%%%
Theta_Ave_left = Theta_Ave_left*180/pi;
Protrusion_angle_left = Protrusion_angle_left*180/pi;

```

```

lambda_left = 360/w_left    %degrees hereon
sorted_angles_left = sort(Protrusion_angle_left);
T_left = []
T_left(1) = sorted_angles_left(2) - sorted_angles_left(1);
for i = 2:(w_left - 1)
    T_left(i) = sorted_angles_left(i + 1) - sorted_angles_left(i);
end
T_left = T_left';
U_argument_left = abs(T_left - lambda_left);
U_left = 1/2*sum(U_argument_left);

Theta_Ave_right = Theta_Ave_right*180/pi;
Protrusion_angle_right = Protrusion_angle_right*180/pi;

T_right = []
lambda_right = 360/w_right    %degrees hereon
sorted_angles_right = sort(Protrusion_angle_right);
T_right(1) = sorted_angles_right(2) - sorted_angles_right(1);
for i = 2:(w_right - 1)
    T_right(i) = sorted_angles_right(i + 1) - sorted_angles_right(i);
end
T_right = T_right';
U_argument_right = abs(T_right - lambda_right);
U_right = 1/2*sum(U_argument_right);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Rayleigh's Test %%%%%%%%%
z_left = w_left*resultant_length_UnitCircle_left^2;
z_right = w_right*resultant_length_UnitCircle_right^2;

P_left = exp(sqrt(1 + 4*w_left + 4*(w_left^2 - (w_left*resultant_length_UnitCircle_left)^2)) - (1 + 2*w_left));
P_right = exp(sqrt(1 + 4*w_right + 4*(w_right^2 - (w_right*resultant_length_UnitCircle_right)^2)) - (1 + 2*w_right));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Confidence Interval for Average Angles on Unit Circle %%%%%%%%%
chi_square = 3.841; %95 percent confidence_interval
if resultant_length_UnitCircle_left <= 0.9 && resultant_length_UnitCircle_left > sqrt(chi_square/(2*w_left))
    d_unit_circle_left = ceil((180/pi)*acos(sqrt((2*w_left*(2*(w_left*resultant_length_UnitCircle_left)^2 - w_left*chi_square))/(4*w_left - chi_square)))/w_left*resultant_length_UnitCircle_left));
    save('Confidence_Interval_UnitCircle_left LATE.txt','d_unit_circle_left','-ascii','-double');
end
if resultant_length_UnitCircle_right <= 0.9 && resultant_length_UnitCircle_right > sqrt(chi_square/(2*w_right))
    d_unit_circle_right = ceil((180/pi)*acos(sqrt((2*w_right*(2*(w_right*resultant_length_UnitCircle_right)^2 - w_right*chi_square))/(4*w_right - chi_square)))/w_right*resultant_length_UnitCircle_right));
    save('Confidence_Interval_UnitCircle_right LATE.txt','d_unit_circle_right','-ascii','-double');
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

x_confidence_interval_UnitCircle_left = t_w_left_minus_1*std_UnitCircle_x_left/sqrt(w_left);
y_confidence_interval_UnitCircle_left = t_w_left_minus_1*std_UnitCircle_y_left/sqrt(w_left);
x_confidence_interval_UnitCircle_right = t_w_right_minus_1*std_UnitCircle_x_right/sqrt(w_right);
y_confidence_interval_UnitCircle_right = t_w_right_minus_1*std_UnitCircle_y_right/sqrt(w_right);

summary_UnitCircle_left = [initial_separation Theta_Ave_left resultant_length_UnitCircle_left w_left P_left U_left
x_UnitCircle_avg_left std_UnitCircle_x_left x_confidence_interval_UnitCircle_left y_UnitCircle_avg_left
std_UnitCircle_y_left y_confidence_interval_UnitCircle_left];

```

```
summary_UnitCircle_right = [initial_separation Theta_Ave_right resultant_length_UnitCircle_right w_right P_right
U_right x_UnitCircle_avg_right std_UnitCircle_x_right x_confidence_interval_UnitCircle_right
y_UnitCircle_avg_right std_UnitCircle_y_right y_confidence_interval_UnitCircle_right]';
```

```
save('Summary UnitCircle Left Late.txt','summary_UnitCircle_left','-ascii','-double');
save('Summary UnitCircle Right Late.txt','summary_UnitCircle_right','-ascii','-double');
save('Critical_Value_for_V-Test(u)_Left Late.txt','u_left','-ascii','-double');
save('Critical_Value_for_V-Test(u)_Right Late.txt','u_right','-ascii','-double');
%save('x-coordinates_protrusions_left.txt','x_UnitCircle_left','-ascii','-double');
%save('y-coordinates_protrusions_left.txt','y_UnitCircle_left','-ascii','-double');
%save('x-coordinates_protrusions_right.txt','x_UnitCircle_right','-ascii','-double');
%save('y-coordinates_protrusions_right.txt','y_UnitCircle_right','-ascii','-double');
```

```
%%%%%%%%%%%%%% %%%%%%%%%%%%%%
%%%%%%%%%%%%%% %%%%%%%%%%%%%%
%%%%%%%%%%%%%% %%%%%%%%%%%%%%
%%%%%%%%%%%%%% %%%%%%%%%%%%%%
```

```
x_left = x_P_double_prime_left_late;
y_left = y_P_double_prime_left_late;
x_right = x_P_double_prime_right_late;
y_right = y_P_double_prime_right_late;
```

```
y_avg_left = mean(y_left);
x_avg_left = mean(x_left);
y_avg_right = mean(y_right);
x_avg_right = mean(x_right);
```

```
std_x_left = std(x_left);
std_y_left = std(y_left);
std_x_right = std(x_right);
std_y_right = std(y_right);
```

```
s1_squared_left = sum((x_left - x_avg_left).^2*(1/(w_left - 1))); % [length^2]
s2_squared_left = sum((y_left - y_avg_left).^2*(1/(w_left - 1))); % [length^2]
s1_squared_right = sum((x_right - x_avg_right).^2*(1/(w_right - 1))); % [length^2]
s2_squared_right = sum((y_right - y_avg_right).^2*(1/(w_right - 1))); % [length^2]
```

```
r_left = sum((x_left - x_avg_left).*(y_left - y_avg_left))/((w_left - 1)*sqrt(s1_squared_left)*sqrt(s2_squared_left));
% [dimensionless]
```

```
T_squared_confidence_left = 2*((w_left - 1)/(w_left - 2))*F_2_w_left_minus_2;
T_squared_left = (w_left/(1 - r_left^2))*(x_avg_left^2/s1_squared_left -
2*r_left*x_avg_left*y_avg_left/(sqrt(s1_squared_left)*sqrt(s2_squared_left)) + y_avg_left^2/s2_squared_left);
```

```
r_right = sum((x_right - x_avg_right).*(y_right - y_avg_right))/((w_right -
1)*sqrt(s1_squared_right)*sqrt(s2_squared_right)); % [dimensionless]
T_squared_confidence_right = 2*((w_right - 1)/(w_right - 2))*F_2_w_right_minus_2;
T_squared_right = (w_right/(1 - r_right^2))*(x_avg_right^2/s1_squared_right -
2*r_right*x_avg_right*y_avg_right/(sqrt(s1_squared_right)*sqrt(s2_squared_right)) +
y_avg_right^2/s2_squared_right);
```

```
A_left = s2_squared_left; % [length^2]
A_right = s2_squared_right; % [length^2]
B_left = -r_left*sqrt(s1_squared_left)*sqrt(s2_squared_left); % [length^2]
B_right = -r_right*sqrt(s1_squared_right)*sqrt(s2_squared_right); % [length^2]
C_left = s1_squared_left; % [length^2]
```

```

C_right = s1_squared_right;    % [length^2]
D_left = (1 - r_left^2)*s1_squared_left*s2_squared_left;    % [length^4]
D_right = (1 - r_right^2)*s1_squared_right*s2_squared_right;    % [length^4]
R_left = sqrt((A_left - C_left)^2 + 4*B_left^2);    % [length^2]
R_right = sqrt((A_right - C_right)^2 + 4*B_right^2);    % [length^2]
a_left = sqrt(2*D_left/(A_left + C_left - R_left));    % [length]
a_right = sqrt(2*D_right/(A_right + C_right - R_right));    % [length]
b_left = sqrt(2*D_left/(A_left + C_left + R_left));    % [length]
b_right = sqrt(2*D_right/(A_right + C_right + R_right));    % [length]
theta_left = atan(2*B_left/(A_left - C_left - R_left));
theta_right = atan(2*B_right/(A_right - C_right - R_right));

H_left = A_left*C_left - B_left^2;
H_right = A_right*C_right - B_right^2;
G_left = A_left*x_avg_left^2 + 2*B_left*x_avg_left*y_avg_left + C_left*y_avg_left^2 - D_left;
G_right = A_right*x_avg_right^2 + 2*B_right*x_avg_right*y_avg_right + C_right*y_avg_right^2 - D_right;
U_left = (H_left*x_avg_left^2 - C_left*D_left)^(-1);
U_right = (H_right*x_avg_right^2 - C_right*D_right)^(-1);
V_left = (D_left*G_left*H_left)^(1/2);
V_right = (D_right*G_right*H_right)^(1/2);
W_left = H_left*x_avg_left*y_avg_left + B_left*D_left;
W_right = H_right*x_avg_right*y_avg_right + B_right*D_right;
m1_left = U_left*(W_left + V_left);
m2_left = U_left*(W_left - V_left);
m1_right = U_right*(W_right + V_right);
m2_right = U_right*(W_right - V_right);
m1_theta_left = (180/pi)*atan(m1_left);
m2_theta_left = (180/pi)*atan(m2_left);
m1_theta_right = (180/pi)*atan(m1_right);
m2_theta_right = (180/pi)*atan(m2_right);

D_confidence_left = (1 - r_left^2)*s1_squared_left*s2_squared_left*(1/w_left)*T_squared_confidence_left;
% [length^4]
a_confidence_left = a_left*(w_left^(-1/2))*sqrt(T_squared_confidence_left);
b_confidence_left = b_left*(w_left^(-1/2))*sqrt(T_squared_confidence_left);
D_confidence_right = (1 -
r_right^2)*s1_squared_right*s2_squared_right*(1/w_right)*T_squared_confidence_right;    % [length^4]
a_confidence_right = a_right*(w_right^(-1/2))*sqrt(T_squared_confidence_right);
b_confidence_right = b_right*(w_right^(-1/2))*sqrt(T_squared_confidence_right);

phi = (1:360)';
phi = phi*(pi/180);
X_left = x_avg_left + a_left*cos(theta_left)*cos(phi) - b_left*sin(theta_left)*sin(phi);
Y_left = y_avg_left + a_left*sin(theta_left)*cos(phi) + b_left*cos(theta_left)*sin(phi);
X_confidence_left = x_avg_left + a_confidence_left*cos(theta_left)*cos(phi) -
b_confidence_left*sin(theta_left)*sin(phi);
Y_confidence_left = y_avg_left + a_confidence_left*sin(theta_left)*cos(phi) +
b_confidence_left*cos(theta_left)*sin(phi);
X_right = x_avg_right + a_right*cos(theta_right)*cos(phi) - b_right*sin(theta_right)*sin(phi);
Y_right = y_avg_right + a_right*sin(theta_right)*cos(phi) + b_right*cos(theta_right)*sin(phi);
X_confidence_right = x_avg_right + a_confidence_right*cos(theta_right)*cos(phi) -
b_confidence_right*sin(theta_right)*sin(phi);
Y_confidence_right = y_avg_right + a_confidence_right*sin(theta_right)*cos(phi) +
b_confidence_right*cos(theta_right)*sin(phi);

```

```

figure()
plot(X_left,Y_left,'.b')
hold on
plot(x_left,y_left,'ro')
plot(X_confidence_left,Y_confidence_left,'.k')
hold off

figure()
plot(X_right,Y_right,'.b')
hold on
plot(x_right,y_right,'ro')
plot(X_confidence_right,Y_confidence_right,'.k')
hold off

x_confidence_interval_left = t_w_left_minus_1*std_x_left/sqrt(w_left);
y_confidence_interval_left = t_w_left_minus_1*std_y_left/sqrt(w_left);
x_confidence_interval_right = t_w_right_minus_1*std_x_right/sqrt(w_right);
y_confidence_interval_right = t_w_right_minus_1*std_y_right/sqrt(w_right);

theta_left = theta_left*180/pi;
theta_right = theta_right*180/pi;

avg_resultant_vector_angle_left = atan(y_avg_left/x_avg_left);
avg_resultant_vector_angle_right = atan(y_avg_right/x_avg_right);

if x_avg_left < 0
    avg_resultant_vector_angle_left = avg_resultant_vector_angle_left + pi;
end
if x_avg_left > 0 && y_avg_left < 0
    avg_resultant_vector_angle_left = avg_resultant_vector_angle_left + 2*pi;
end
if x_avg_left == 0 && y_avg_left < 0
    avg_resultant_vector_angle_left = 3*pi/2;
end
if x_avg_left == 0 && y_avg_left > 0
    avg_resultant_vector_angle_left = pi/2;
end

if x_avg_right < 0
    avg_resultant_vector_angle_right = avg_resultant_vector_angle_right + pi;
end
if x_avg_right > 0 && y_avg_right < 0
    avg_resultant_vector_angle_right = avg_resultant_vector_angle_right + 2*pi;
end
if x_avg_right == 0 && y_avg_right < 0
    avg_resultant_vector_angle_right = 3*pi/2;
end
if x_avg_right == 0 && y_avg_right > 0
    avg_resultant_vector_angle_right = pi/2;
end

avg_resultant_vector_angle_left = (180/pi)*avg_resultant_vector_angle_left;
avg_resultant_vector_angle_right = (180/pi)*avg_resultant_vector_angle_right;
mean_vector_magnitude_left = mean(sqrt(x_left.^2 + y_left.^2));
mean_vector_magnitude_right = mean(sqrt(x_right.^2 + y_right.^2));

```



```

protrusion_frequency_left = w_left/((max(Protrusions_slice_left)-72)*(12/60));
protrusion_frequency_right = w_right/((max(Protrusions_slice_right)-72)*(12/60));

summary_left = [initial_separation theta_left avg_resultant_vector_angle_left T_squared_left
T_squared_confidence_left a_left w_left m1_theta_left m2_theta_left mean_vector_magnitude_left x_avg_left
std_x_left x_confidence_interval_left y_avg_left std_y_left y_confidence_interval_left protrusion_frequency_left];
summary_right = [initial_separation theta_right avg_resultant_vector_angle_right T_squared_right
T_squared_confidence_right a_right w_right m1_theta_right m2_theta_right mean_vector_magnitude_right
x_avg_right std_x_right x_confidence_interval_right y_avg_right std_y_right y_confidence_interval_right
protrusion_frequency_right]';

save('x-coordinates_confidence_ellipse_LEFT_Late.txt','X_confidence_left','-ascii','-double');
save('y-coordinates_confidence_ellipse_LEFT_Late.txt','Y_confidence_left','-ascii','-double');
save('x-coordinates_confidence_ellipse_RIGHT_Late.txt','X_confidence_right','-ascii','-double');
save('y-coordinates_confidence_ellipse_RIGHT_Late.txt','Y_confidence_right','-ascii','-double');
save('Summary_Left_Late.txt','summary_left','-ascii','-double');
save('Summary_Right_Late.txt','summary_right','-ascii','-double');
%save('x-coordinates_protrusions_left.txt','x_UnitCircle_left','-ascii','-double');
%save('y-coordinates_protrusions_left.txt','y_UnitCircle_left','-ascii','-double');
%save('x-coordinates_protrusions_right.txt','x_UnitCircle_right','-ascii','-double');
%save('y-coordinates_protrusions_right.txt','y_UnitCircle_right','-ascii','-double');

%%%%%%%%%%%%%% Binning %%%%%%%%%%%%%%%

%BIN_10_left = hist(Protrusion_angle_left,36);
%BIN_10_left = BIN_10_left';

save('Protrusion_Histogram_Bin30_Left_Late.txt','Binned_protrusions_30_left_late','-ascii','-double');
save('Protrusion_Histogram_Bin180_Left_Late.txt','Binned_protrusions_180_left_late','-ascii','-double');
save('Protrusion_Histogram_Bin30_Right_Late.txt','Binned_protrusions_30_right_late','-ascii','-double');
save('Protrusion_Histogram_Bin180_Right_Late.txt','Binned_protrusions_180_right_late','-ascii','-double');

save('x-Coordinate_Protrusion_Lenghts_from_Cell_Membrane_Left_Late.txt','x_P_double_prime_left_late','-
ascii','-double');
save('y-Coordinate_Protrusion_Lenghts_from_Cell_Membrane_Left_Late.txt','y_P_double_prime_left_late','-
ascii','-double');
save('x-Coordinate_Protrusion_Lenghts_from_Cell_Membrane_Right_Late.txt','x_P_double_prime_right_late','-
ascii','-double');
save('y-Coordinate_Protrusion_Lenghts_from_Cell_Membrane_Right_Late.txt','y_P_double_prime_right_late','-
ascii','-double');

```

D-3 Protrusion dynamics of cell and boundary

```

%ADS 12-27-2010 (modified)
%This code will carry out the data analysis of circular statistics for the
%Cell-search Experiments
% It carries out V-Test for each set of angular data as well as Rao's Test

```

% Spacing Test.

%This section will read data from Excel file

clear

clc

sprintf('%s','Lets roll dude!')

%%%%%%%%%%%%%% Cell L

%%%%%%%%%%%%%%

sprintf('%s','Select LEFT CELL Data')

LEFT = xlsread('C:\Users\Anthony Diaz-Santana\Documents\MATLAB\Position1_Data_cell_boundary.xls',-1);

Uncorrected_cell_center_x_left = LEFT(:,1);

Uncorrected_cell_center_y_left = LEFT(:,2);

Uncorrected_slicee_left = LEFT(:,5);

Uncorrected_slice_left = round(Uncorrected_slicee_left/0.3125);

sprintf('%s','Select REFERENCE Data')

REFERENCE = xlsread('C:\Users\Anthony Diaz-

Santana\Documents\MATLAB\Position1_Data_cell_boundary.xls',-1);

if REFERENCE ~= 0

Ref_x = REFERENCE(:,1);

Ref_y = REFERENCE(:,2);

Ref_slicee = REFERENCE(:,5);

if Ref_slicee ~= 0

Ref_slice = round(Ref_slicee/0.3125);

end

else

Ref_slicee = 0;

end

sprintf('%s','Select LEFT CELL PROTRUSION Data')

PROTRUSION_LEFT = xlsread('C:\Users\Anthony Diaz-

Santana\Documents\MATLAB\Position1_Data_cell_boundary.xls',-1);

Uncorrected_Protrusions_x_left = PROTRUSION_LEFT(:,1);

Uncorrected_Protrusions_y_left = PROTRUSION_LEFT(:,2);

Protrusions_slicee_left = PROTRUSION_LEFT(:,5);

Protrusions_slice_left = round(Protrusions_slicee_left/0.3125);

interval = input('What is the time interval used (in minutes)? _ ');

%cell_area_left = input('What is the area of the LEFT cell in square microns? _ ');

%cell_radius_left = sqrt(cell_area_left/pi);

%protrusion_threshold = input('What is the protrusion threshold in microns you want to use? _ ');

%%%%%%%%%%%%%% Cell R

%%%%%%%%%%%%%%

sprintf('%s','Select BOUNDARY Data')

RIGHT = xlsread('C:\Users\Anthony Diaz-Santana\Documents\MATLAB\Position1_Data_cell_boundary.xls',-1);

Uncorrected_cell_center_x_right = RIGHT(:,1);

Uncorrected_cell_center_y_right = RIGHT(:,2);

Uncorrected_slicee_right = RIGHT(:,5);

Uncorrected_slice_right = round(Uncorrected_slicee_right/0.3125);

%%%%%%%%%%%%%%

%%%%%%%%%%%%%%

```

i = length(Uncorrected_cell_center_x_right);
w_left = length(Uncorrected_Protrusions_y_left);

delta_Ref_x = zeros(i,1);
delta_Ref_x(1,1) = 0;
delta_Ref_y = zeros(i,1);
delta_Ref_y(1,1) = 0;

deltaUncorrected_cell_center_x_right = zeros(i,1);
deltaUncorrected_cell_center_y_right = zeros(i,1);
deltaUncorrected_cell_center_x_right(1,1) = 0;
deltaUncorrected_cell_center_y_right(1,1) = 0;

deltaUncorrected_cell_center_x_left = zeros(i,1);
deltaUncorrected_cell_center_y_left = zeros(i,1);
deltaUncorrected_cell_center_x_left(1,1) = 0;
deltaUncorrected_cell_center_y_left(1,1) = 0;

deltaUncorrected_protrusion_x_left = zeros(w_left,1);
deltaUncorrected_protrusion_y_left = zeros(w_left,1);
deltaUncorrected_protrusion_x_left(1,1) = 0;
deltaUncorrected_protrusion_y_left(1,1) = 0;

time_min = 0:interval:i*interval;
time = time_min/60;
time_in_hours = i*interval/60;

if Ref_slicee ~= 0
    for j = 2:i
        delta_Ref_x(j,1) = Ref_x(j,1) - Ref_x(j - 1,1);
        delta_Ref_y(j,1) = Ref_y(j,1) - Ref_y(j - 1,1);
        deltaUncorrected_cell_center_x_left(j,1) = Uncorrected_cell_center_x_left(j,1) -
Uncorrected_cell_center_x_left(j - 1,1);
        deltaUncorrected_cell_center_y_left(j,1) = Uncorrected_cell_center_y_left(j,1) -
Uncorrected_cell_center_y_left(j - 1,1);
        deltaUncorrected_cell_center_x_right(j,1) = Uncorrected_cell_center_x_right(j,1) -
Uncorrected_cell_center_x_right(j - 1,1);
        deltaUncorrected_cell_center_y_right(j,1) = Uncorrected_cell_center_y_right(j,1) -
Uncorrected_cell_center_y_right(j - 1,1);
    end
    c_cell_center_x_left = Uncorrected_cell_center_x_left + (deltaUncorrected_cell_center_x_left - delta_Ref_x);
    c_cell_center_y_left = Uncorrected_cell_center_y_left + (deltaUncorrected_cell_center_y_left - delta_Ref_y);
    c_cell_center_x_right = Uncorrected_cell_center_x_right + (deltaUncorrected_cell_center_x_right -
delta_Ref_x);
    c_cell_center_y_right = Uncorrected_cell_center_y_right + (deltaUncorrected_cell_center_y_right -
delta_Ref_y);
else
    c_cell_center_x_left = Uncorrected_cell_center_x_left;
    c_cell_center_y_left = Uncorrected_cell_center_y_left;
    c_cell_center_x_right = Uncorrected_cell_center_x_right;
    c_cell_center_y_right = Uncorrected_cell_center_y_right;
end

if Ref_slicee == 0

```

```

    for j = 2:w_left
        deltaUncorrected_protrusion_x_left(j,1) = Uncorrected_Protrusions_x_left(j,1) -
        Uncorrected_Protrusions_x_left(j - 1,1);
        deltaUncorrected_protrusion_y_left(j,1) = Uncorrected_Protrusions_y_left(j,1) -
        Uncorrected_Protrusions_y_left(j - 1,1);
        c_Protrusions_x_left(j,1) = Uncorrected_Protrusions_x_left(j,1) + (deltaUncorrected_protrusion_x_left(j,1) -
        delta_Ref_x(Protrusions_slice_left(j)));
        c_Protrusions_y_left(j,1) = Uncorrected_Protrusions_y_left(j,1) + (deltaUncorrected_protrusion_y_left(j,1) -
        delta_Ref_y(Protrusions_slice_left(j)));
    end
    c_Protrusions_x_left(1,1) = Uncorrected_Protrusions_x_left(1,1) + (deltaUncorrected_protrusion_x_left(1,1) -
    delta_Ref_x(Protrusions_slice_left(1)));
    c_Protrusions_y_left(1,1) = Uncorrected_Protrusions_y_left(1,1) + (deltaUncorrected_protrusion_y_left(1,1) -
    delta_Ref_y(Protrusions_slice_left(1)));
    else
        c_Protrusions_x_left = Uncorrected_Protrusions_x_left;
        c_Protrusions_y_left = Uncorrected_Protrusions_y_left;
    end

    %This section will assign a protrusion to its corresponding slice/time and
    %calculate protrusion length-coordinates

    for m = 1:w_left
        c_Protrusion_lengths_x_left(m,1) = c_Protrusions_x_left(m) - c_cell_center_x_left(Protrusions_slice_left(m));
        c_Protrusion_lengths_y_left(m,1) = c_Protrusions_y_left(m) - c_cell_center_y_left(Protrusions_slice_left(m));
    end

    %%%%%%%%% Cell Reference (for Delta Theta) %%%%%%%%%

    delta_RL_x = c_cell_center_x_right - c_cell_center_x_left;
    delta_RL_y = c_cell_center_y_right - c_cell_center_y_left;

    %%%%%%%%% Delta Theta Left Cell
    %%%%%%%%%

    mag_deltaRL = sqrt(delta_RL_x.^2 + delta_RL_y.^2);
    mag_c_Protrusion_lengths_left = sqrt(c_Protrusion_lengths_x_left.^2 + c_Protrusion_lengths_y_left.^2);

    for m = 1:w_left
        dot_left(m,1) = delta_RL_x(Protrusions_slice_left(m))*c_Protrusion_lengths_x_left(m,1) +
        delta_RL_y(Protrusions_slice_left(m))*c_Protrusion_lengths_y_left(m,1);
        Protrusion_delta_Theta_left(m,1) =
        acos(dot_left(m,1)/(mag_c_Protrusion_lengths_left(m,1)*mag_deltaRL(Protrusions_slice_left(m))));
    end

    %%%%%%%%%
    %%%%%%%%%
    %%%%%%%%%

    %%%%%%%%% Calculating angle of protrusion in the appropriate coordinate
    %%%%%%%%% system and correcting for quadrant location

    Protrusion_angle_left = atan(y_P_double_prime_left./x_P_double_prime_left);
    Protrusions_Unit_Circle_x_left = cos(Protrusion_angle_left);

```

```

Protrusions_Unit_Circle_y_left = sin(Protrusion_angle_left);

for m = 1:w_left

    if x_P_double_prime_left(m,1) < 0
        Protrusion_angle_left(m,1) = Protrusion_angle_left(m,1) + pi;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
    if x_P_double_prime_left(m,1) > 0 && y_P_double_prime_left(m,1) < 0
        Protrusion_angle_left(m,1) = Protrusion_angle_left(m,1) + 2*pi;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
    if x_P_double_prime_left(m,1) == 0 && y_P_double_prime_left(m,1) < 0
        Protrusion_angle_left(m,1) = 3*pi/2;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
    if x_P_double_prime_left(m,1) == 0 && y_P_double_prime_left(m,1) > 0
        Protrusion_angle_left(m,1) = pi/2;
        Protrusions_Unit_Circle_x_left(m,1) = cos(Protrusion_angle_left(m,1));
        Protrusions_Unit_Circle_y_left(m,1) = sin(Protrusion_angle_left(m,1));
    end
end

Protrusion_lengths_from_membrane_x_left = x_P_double_prime_left; % -
cell_radius_left*cos(Protrusion_delta_Theta_left(m,1));
Protrusion_lengths_from_membrane_y_left = y_P_double_prime_left; % -
cell_radius_left*sin(Protrusion_delta_Theta_left(m,1));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%% Resultant Lenght for Set of Angles on Unit Circle %%%%%%%%%

X_left = (1/w_left)*sum(Protrusions_Unit_Circle_x_left);
Y_left = (1/w_left)*sum(Protrusions_Unit_Circle_y_left);

r_left = sqrt(X_left^2 + Y_left^2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%% Resultant Length for Set of Angles Length-based %%%%%%%%%

X_length_left = (1/w_left)*sum(Protrusion_lengths_from_membrane_x_left);
Y_length_left = (1/w_left)*sum(Protrusion_lengths_from_membrane_y_left);

r_length_left = sqrt(X_length_left^2 + Y_length_left^2);

r1_left = sqrt(Protrusion_lengths_from_membrane_x_left.^2 + Protrusion_lengths_from_membrane_y_left.^2);
r_max_left = max(r1_left);
r_mean_left = mean(r1_left);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Moore modified Rayleigh
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

vectors_left = [r1_left Protrusion_angle_left];
sorted_lengths_from_membrane_left = sortrows(vectors_left);
n_left = (1:w_left)';
X_Moore_left = sum(n_left.*cos(vectors_left(:,2)));
Y_Moore_left = sum(n_left.*sin(vectors_left(:,2)));
R_Moore_left = sqrt(X_Moore_left^2 + Y_Moore_left^2);
R_critical_Moore_left = R_Moore_left/(w_left^(3/2));
Double_Tail_R_critical_left = 2*R_critical_Moore_left;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Parametric One-Sample Second-Order Analysis %%%%%%%%%%
normalized_r1_left = r1_left/max(r1_left);
normalized_Protrusion_lengths_from_membrane_x_left = normalized_r1_left.*cos(Protrusion_angle_left);
normalized_Protrusion_lengths_from_membrane_y_left = normalized_r1_left.*sin(Protrusion_angle_left);
normalized_X_length_left = (1/w_left)*sum(normalized_Protrusion_lengths_from_membrane_x_left);
normalized_Y_length_left = (1/w_left)*sum(normalized_Protrusion_lengths_from_membrane_y_left);
sum_x_square_left = sum(normalized_Protrusion_lengths_from_membrane_x_left.^2) -
(1/w_left)*(sum(normalized_Protrusion_lengths_from_membrane_x_left))^2;
sum_y_square_left = sum(normalized_Protrusion_lengths_from_membrane_y_left.^2) -
(1/w_left)*(sum(normalized_Protrusion_lengths_from_membrane_y_left))^2;
sum_xy_left =
sum(normalized_Protrusion_lengths_from_membrane_x_left.*normalized_Protrusion_lengths_from_membrane_y_l
eft) -
(1/w_left)*(sum(normalized_Protrusion_lengths_from_membrane_x_left)*(sum(normalized_Protrusion_lengths_fro
m_membrane_y_left)));
F_left = (w_left/2)*(w_left - 2)*(normalized_X_length_left^2*sum_y_square_left -
2*normalized_X_length_left*normalized_Y_length_left*sum_xy_left +
normalized_Y_length_left^2*sum_x_square_left)/(sum_x_square_left*sum_y_square_left - sum_xy_left^2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Binning %%%%%%%%%%

BIN_10_left = hist(Protrusion_angle_left,36);
BIN_180_left = hist(Protrusion_angle_left,2);
BIN_30_left = hist(Protrusion_angle_left,12);

BIN_10_left = BIN_10_left';
BIN_180_left = BIN_180_left';
BIN_30_left = BIN_30_left';

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Average Angle Calculations (Length Weighted and Unit Circle) %%%%%%%%%%

length_weighted_avg_angle_left = atan(Y_length_left/X_length_left);

if X_length_left < 0
    length_weighted_avg_angle_left = length_weighted_avg_angle_left + pi;
end
if X_length_left > 0 && Y_length_left < 0
    length_weighted_avg_angle_left = length_weighted_avg_angle_left + 2*pi;
end
if X_length_left == 0 && Y_length_left < 0
    length_weighted_avg_angle_left = 3*pi/2;
end
if X_length_left == 0 && Y_length_left > 0
    length_weighted_avg_angle_left = pi/2;
end

```

```

Theta_Ave_left = atan(Y_left/X_left);

if X_left < 0
    Theta_Ave_left = Theta_Ave_left + pi;
end
if X_left > 0 && Y_left < 0
    Theta_Ave_left = Theta_Ave_left + 2*pi;
end
if X_left == 0 && Y_left < 0
    Theta_Ave_left = 3*pi/2;
end
if X_left == 0 && Y_left > 0
    Theta_Ave_left = pi/2;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Confidence Interval for Average Angles %%%%%%%%%
chi_square = 3.841; %95 percent confidence_interval
if r_left <= 0.9 && r_left > sqrt(chi_square/(2*w_left))
    d_unit_circle_left = ceil(acos(sqrt((2*w_left*(2*(w_left*r_left)^2 - w_left*chi_square)/(4*w_left -
chi_square)))/w_left*r_left)*180/pi);
end
if r_left > 0.9
    d_unit_circle_left = ceil(acos(sqrt(w_left^2 - (w_left^2 -
(w_left*r_left)^2)*exp(chi_square/w_left))/(w_left*r_left)));
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% V-Test %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for phi = 1:360
    V_left(phi) = w_left*r_left*cos(Theta_Ave_left - phi*pi/180);
    u_left(phi) = V_left(phi)*sqrt(2/w_left);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Rao's Spacing Test %%%%%%%%%
length_weighted_avg_angle_left = length_weighted_avg_angle_left*180/pi;
Theta_Ave_left = Theta_Ave_left*180/pi;
Protrusion_angle_left = Protrusion_angle_left*180/pi;

lambda_left = 360/w_left; %degrees hereon
sorted_angles_left = sort(Protrusion_angle_left);
T_left(1) = sorted_angles_left(2) - sorted_angles_left(1);
for i = 2:(w_left - 1)
    T_left(i) = sorted_angles_left(i + 1) - sorted_angles_left(i);
end
T_left = T_left';
U_argument_left = abs(T_left - lambda_left);
U_left = 1/2*sum(U_argument_left);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Rayleigh's Test %%%%%%%%%
z_left = w_left*r_left^2;
P_left = exp(sqrt(1 + 4*w_left + 4*(w_left^2 - (w_left*r_left)^2)) - (1 + 2*w_left));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Square displacement %%%%%%%%%

```

```

MSD_from_Origin_left = (c_cell_center_x_left(:,1) - c_cell_center_x_left(1,1)).^2 + (c_cell_center_y_left(:,1) -
c_cell_center_y_left(1,1)).^2;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

summary_left = [Theta_Ave_left length_weighted_avg_angle_left r_mean_left r_max_left w_left z_left P_left
U_left F_left r_left];
V_test_left = V_left;
u_left = u_left;
coordinates_left = [Protrusions_Unit_Circle_x_left Protrusions_Unit_Circle_y_left
Protrusion_lengths_from_membrane_x_left Protrusion_lengths_from_membrane_y_left];

save('Time_in_Hours.txt', 'time', '-ascii', '-double');
save('Square_Displacement_from_Origin_Left.txt', 'MSD_from_Origin_left', '-ascii', '-double');
save('SummaryMatrix_Left.txt', 'summary_left', '-ascii', '-double');
save('Unit_Circle_x_Left.txt', 'Protrusions_Unit_Circle_x_left', '-ascii', '-double');
save('Unit_Circle_y_Left.txt', 'Protrusions_Unit_Circle_y_left', '-ascii', '-double');
save('x-
Coordinate_Protrusion_Lenghts_from_Cell_Membrane_Left.txt', 'Protrusion_lengths_from_membrane_x_left', '-
ascii', '-double');
save('y-
Coordinate_Protrusion_Lenghts_from_Cell_Membrane_Left.txt', 'Protrusion_lengths_from_membrane_y_left', '-
ascii', '-double');
save('Critical_Value_for_V-Test(u)_Left.txt', 'u_left', '-ascii', '-double');
save('Protrusion_Histogram_Bin10_Left.txt', 'BIN_10_left', '-ascii', '-double');
save('Protrusion_Histogram_Bin180_Left.txt', 'BIN_180_left', '-ascii', '-double');
save('Protrusion_Histogram_Bin30_Left.txt', 'BIN_30_left', '-ascii', '-double');
save('Confidence_Interval_for_AverageAngle_95%Confidence.txt', 'd_unit_circle', '-ascii', '-double');

```


D-4 Protrusion Dynamics of single cells

```
%ADS 12-27-2010 (modified)
%This code will carry out the data analysis of circular statistics for the
%Cell-search Experiments
% It carries out V-Test for each set of angular data as well as Rao's Test
% Spacing Test.

%This section will read data from Excel file
clear
clc
sprintf('%s','Lets roll dude!')

sprintf('%s','Select CELL CENTER Data')
CELL = xlsread('C:\Users\Anthony Diaz-Santana\Documents\MATLAB\Position1_Data.xls',-1);
Uncorrected_cell_center_x = CELL(:,1);
Uncorrected_cell_center_y = CELL(:,2);
Uncorrected_slicee = CELL(:,5);
Uncorrected_slice = round(Uncorrected_slicee/0.3125);

sprintf('%s','Select REFERENCE Data')
REFERENCE = xlsread('C:\Users\Anthony Diaz-Santana\Documents\MATLAB\Position1_Data.xls',-1);
if REFERENCE ~= 0
    Ref_x = REFERENCE(:,1);
    Ref_y = REFERENCE(:,2);
    Ref_slicee = REFERENCE(:,5);
    if Ref_slicee ~= 0
        Ref_slice = round(Ref_slicee/0.3125);
    end
else
    Ref_slicee = 0;
end

sprintf('%s','Select CELL PROTRUSION Data')
PROTRUSIONS = xlsread('C:\Users\Anthony Diaz-Santana\Documents\MATLAB\Position1_Data.xls',-1);
Uncorrected_Protrusions_x = PROTRUSIONS(:,1);
Uncorrected_Protrusions_y = PROTRUSIONS(:,2);
Protrusions_slicee = PROTRUSIONS(:,5);
Protrusions_slice = round(Protrusions_slicee/0.3125);

interval = input('What is the time interval used (in minutes)? _ ');
cell_area = input('What is the area of this cell in square microns? _ ');
cell_radius = sqrt(cell_area/pi);
%protrusion_threshold = input('What is the protrusion threshold in microns you want to use? _ ');

i = length(Uncorrected_cell_center_x);
w = length(Uncorrected_Protrusions_x);

delta_Ref_x = zeros(i,1);
```

```

delta_Ref_x(1,1) = 0;
delta_Ref_y = zeros(i,1);
delta_Ref_y(1,1) = 0;
deltaUncorrected_cell_center_x = zeros(i,1);
deltaUncorrected_cell_center_y = zeros(i,1);
deltaUncorrected_cell_center_x(1,1) = 0;
deltaUncorrected_cell_center_y(1,1) = 0;
time_min = 0:interval:i*interval;
time = time_min/60';
Binned_protrusions_360 = zeros(12,1);
%Binned_protrusions_180 = zeros(6,1);
time_in_hours = i*interval/60;
Protrusion_Histogram = zeros(i,1);
Protrusion_vectors_from_membrane_x = zeros(i,1);
Protrusion_vectors_from_membrane_y = zeros(i,1);
Protrusion_unit_vectors_x = zeros(i,1);
Protrusion_unit_vectors_y = zeros(i,1);

if Ref_slicee ~= 0
    for j = 2:i
        delta_Ref_x(j,1) = Ref_x(j,1) - Ref_x(j - 1,1);
        delta_Ref_y(j,1) = Ref_y(j,1) - Ref_y(j - 1,1);
        deltaUncorrected_cell_center_x(j,1) = Uncorrected_cell_center_x(j,1) - Uncorrected_cell_center_x(j - 1,1);
        deltaUncorrected_cell_center_y(j,1) = Uncorrected_cell_center_y(j,1) - Uncorrected_cell_center_y(j - 1,1);
    end
    c_cell_center_x = Uncorrected_cell_center_x + (deltaUncorrected_cell_center_x - delta_Ref_x);
    c_cell_center_y = Uncorrected_cell_center_y + (deltaUncorrected_cell_center_y - delta_Ref_y);
else
    c_cell_center_x = Uncorrected_cell_center_x;
    c_cell_center_y = Uncorrected_cell_center_y;
end

save('Corrected_Cell_Center_x.txt','c_cell_center_x','-ascii','-double');
save('Corrected_Cell_Center_y.txt','c_cell_center_y','-ascii','-double');

if Ref_slicee ~= 0
    for j = 2:w
        deltaUncorrected_protrusion_x(j,1) = Uncorrected_Protrusions_x(j,1) - Uncorrected_Protrusions_x(j - 1,1);
        deltaUncorrected_protrusion_y(j,1) = Uncorrected_Protrusions_y(j,1) - Uncorrected_Protrusions_y(j - 1,1);
        c_Protrusions_x(j,1) = Uncorrected_Protrusions_x(j,1) + (deltaUncorrected_protrusion_x(j,1) -
delta_Ref_x(Protrusions_slice(j)));
        c_Protrusions_y(j,1) = Uncorrected_Protrusions_y(j,1) + (deltaUncorrected_protrusion_y(j,1) -
delta_Ref_y(Protrusions_slice(j)));
    end
    c_Protrusions_x(1,1) = Uncorrected_Protrusions_x(1,1) + (deltaUncorrected_protrusion_x(1,1) -
delta_Ref_x(Protrusions_slice(1)));
    c_Protrusions_y(1,1) = Uncorrected_Protrusions_y(1,1) + (deltaUncorrected_protrusion_y(1,1) -
delta_Ref_y(Protrusions_slice(1)));
else
    c_Protrusions_x = Uncorrected_Protrusions_x;
    c_Protrusions_y = Uncorrected_Protrusions_y;
end

%This section will assign a protrusion to its corresponding slice/time and
%calculate protrusion length-coordinates

```

```

for m = 1:w
    Protrusion_lengths_x(m,1) = c_Protrusions_x(m) - c_cell_center_x(Protrusions_slice(m));
    Protrusion_lengths_y(m,1) = c_Protrusions_y(m) - c_cell_center_y(Protrusions_slice(m));

    for q = 1:i
        if q == Protrusions_slice(m)
            Protrusion_Histogram(q,1) = Protrusion_Histogram(q,1) + 1;
        end
    end
end

%%%%%% Protrusion angles and unit circle generation %%%%%%%
Protrusion_angle = atan(Protrusion_lengths_y./Protrusion_lengths_x);

Protrusions_Unit_Circle_x = cos(Protrusion_angle);
Protrusions_Unit_Circle_y = sin(Protrusion_angle);

for m = 1:w
    if Protrusion_lengths_x(m,1) < 0
        Protrusion_angle(m,1) = Protrusion_angle(m,1) + pi;
        Protrusions_Unit_Circle_x(m,1) = cos(Protrusion_angle(m,1));
        Protrusions_Unit_Circle_y(m,1) = sin(Protrusion_angle(m,1));
    end
    if Protrusion_lengths_x(m,1) > 0 && Protrusion_lengths_y(m,1) < 0
        Protrusion_angle(m,1) = Protrusion_angle(m,1) + 2*pi;
        Protrusions_Unit_Circle_x(m,1) = cos(Protrusion_angle(m,1));
        Protrusions_Unit_Circle_y(m,1) = sin(Protrusion_angle(m,1));
    end
    if Protrusion_lengths_x(m,1) == 0 && Protrusion_lengths_y(m,1) < 0
        Protrusion_angle(m,1) = 3*pi/2;
        Protrusions_Unit_Circle_x(m,1) = cos(Protrusion_angle(m,1));
        Protrusions_Unit_Circle_y(m,1) = sin(Protrusion_angle(m,1));
    end
    if Protrusion_lengths_x(m,1) == 0 && Protrusion_lengths_y(m,1) > 0
        Protrusion_angle(m,1) = pi/2;
        Protrusions_Unit_Circle_x(m,1) = cos(Protrusion_angle(m,1));
        Protrusions_Unit_Circle_y(m,1) = sin(Protrusion_angle(m,1));
    end
end
figure()
circ_plot(Protrusion_angle);
title('Protrusion Angles');

Protrusion_lengths_from_membrane_x = Protrusion_lengths_x - cell_radius*cos(Protrusion_angle);
Protrusion_lengths_from_membrane_y = Protrusion_lengths_y - cell_radius*sin(Protrusion_angle);

persistence_plot_time = Protrusions_slice*interval/60;
plot(persistence_plot_time, Protrusion_angle*180/pi,'or')
save('Time of Protrusions for Persistence Plot.txt','persistence_plot_time','-ascii','-double');
%%%%%% Resultant Lenght for Set of Angles on Unit Circle %%%%%%%

X = (1/w)*sum(Protrusions_Unit_Circle_x);
Y = (1/w)*sum(Protrusions_Unit_Circle_y);

```

```

r = sqrt(X^2 + Y^2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Resultant Length for Set of Angles Length-based %%%%%%%%%

X_length = (1/w)*sum(Protrusion_lengths_from_membrane_x);
Y_length = (1/w)*sum(Protrusion_lengths_from_membrane_y);

r_length = sqrt(X_length^2 + Y_length^2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% {
r1 = sqrt(Protrusion_lengths_from_membrane_x.^2 + Protrusion_lengths_from_membrane_y.^2);
[rows,cols] = find(r1 <= protrusion_threshold);
for z = 1:length(cols)
    Protrusion_lengths_from_membrane_x(cols(z)) = [];
    Protrusions_Unit_Circle_x(cols(z)) = [];
    Protrusion_lengths_from_membrane_y(cols(z)) = [];
    Protrusions_Unit_Circle_y(cols(z)) = [];
    Protrusion_Histogram_after_threshold((Protrusions_slice(cols(z))),1) =
Protrusion_Histogram((Protrusions_slice(cols(z))),1) - 1;
    Protrusions_slice(cols(z)) = [];
    Protrusion_angle(cols(z)) = [];
end
% }

r1 = sqrt(Protrusion_lengths_from_membrane_x.^2 + Protrusion_lengths_from_membrane_y.^2);
r_max = max(r1);
r_mean = mean(r1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Moore modified Rayleigh
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
vectors = [r1 Protrusion_angle];
sorted_lengths_from_membrane = sortrows(vectors);
n = (1:w)';
X_Moore = sum(n.*cos(vectors(:,2)));
Y_Moore = sum(n.*sin(vectors(:,2)));
R_Moore = sqrt(X_Moore^2 + Y_Moore^2);
R_critical_Moore = R_Moore/(w^(3/2));
Double_Tail_R_critical = 2*R_critical_Moore;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Parametric One-Sample Second-Order Analysis %%%%%%%%%
normalized_r1 = r1/max(r1);
normalized_Protrusion_lengths_from_membrane_x = normalized_r1.*cos(Protrusion_angle);
normalized_Protrusion_lengths_from_membrane_y = normalized_r1.*sin(Protrusion_angle);
normalized_X_length = (1/w)*sum(normalized_Protrusion_lengths_from_membrane_x);
normalized_Y_length = (1/w)*sum(normalized_Protrusion_lengths_from_membrane_y);
sum_x_square = sum(normalized_Protrusion_lengths_from_membrane_x.^2) -
(1/w)*(sum(normalized_Protrusion_lengths_from_membrane_x))^2;
sum_y_square = sum(normalized_Protrusion_lengths_from_membrane_y.^2) -
(1/w)*(sum(normalized_Protrusion_lengths_from_membrane_y))^2;
sum_xy =
sum(normalized_Protrusion_lengths_from_membrane_x.*normalized_Protrusion_lengths_from_membrane_y) -

```

```
(1/w)*(sum(normalized_Protrusion_lengths_from_membrane_x)*(sum(normalized_Protrusion_lengths_from_membrane_y))));
```

```
F = (w/2)*(w - 2)*(normalized_X_length^2*sum_y_square -  
2*normalized_X_length*normalized_Y_length*sum_xy +  
normalized_Y_length^2*sum_x_square)/(sum_x_square*sum_y_square - sum_xy^2);
```

```
%%%%%%%%%%%%%% Binning 30 Degrees Arc %%%%%%%%%%%%%%%  
for m = 1:w
```

```
    if 0 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= pi/6  
        Binned_protrusions_360(1) = Binned_protrusions_360(1) + 1;  
    end
```

```
    if pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= pi/3  
        Binned_protrusions_360(2) = Binned_protrusions_360(2) + 1;  
    end
```

```
    if pi/3 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= pi/2  
        Binned_protrusions_360(3) = Binned_protrusions_360(3) + 1;  
    end
```

```
    if pi/2 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 2*pi/3  
        Binned_protrusions_360(4) = Binned_protrusions_360(4) + 1;  
    end
```

```
    if 2*pi/3 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 5*pi/6  
        Binned_protrusions_360(5) = Binned_protrusions_360(5) + 1;  
    end
```

```
    if 5*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= pi  
        Binned_protrusions_360(6) = Binned_protrusions_360(6) + 1;  
    end
```

```
    if pi < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 7*pi/6  
        Binned_protrusions_360(7) = Binned_protrusions_360(7) + 1;  
    end
```

```
    if 7*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 8*pi/6  
        Binned_protrusions_360(8) = Binned_protrusions_360(8) + 1;  
    end
```

```
    if 8*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 9*pi/6  
        Binned_protrusions_360(9) = Binned_protrusions_360(9) + 1;  
    end
```

```
    if 9*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 10*pi/6  
        Binned_protrusions_360(10) = Binned_protrusions_360(10) + 1;  
    end
```

```
    if 10*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 11*pi/6  
        Binned_protrusions_360(11) = Binned_protrusions_360(11) + 1;  
    end
```

```
    if 11*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 2*pi
```

```

        Binned_protrusions_360(12) = Binned_protrusions_360(12) + 1;
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Binning %%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Average Angle Calculations (Length Weighted and Unit Circle) %%%%%%%%%%

length_weighted_avg_angle = atan(Y_length/X_length);

if X_length < 0
    length_weighted_avg_angle = length_weighted_avg_angle + pi;
end
if X_length > 0 && Y_length < 0
    length_weighted_avg_angle = length_weighted_avg_angle + 2*pi;
end
if X_length == 0 && Y_length < 0
    length_weighted_avg_angle = 3*pi/2;
end
if X_length == 0 && Y_length > 0
    length_weighted_avg_angle = pi/2;
end

Theta_Ave = atan(Y/X);

if X < 0
    Theta_Ave = Theta_Ave + pi;
end
if X > 0 && Y < 0
    Theta_Ave = Theta_Ave + 2*pi;
end
if X == 0 && Y < 0
    Theta_Ave = 3*pi/2;
end
if X == 0 && Y > 0
    Theta_Ave = pi/2;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Confidence Interval for Average Angles %%%%%%%%%%%%%%
chi_square = 3.841; %95 percent confidence_interval
if r <= 0.9 && r > sqrt(chi_square/(2*w))
    d_unit_circle = ceil(acos(sqrt((2*w*(2*(w*r)^2 - w*chi_square))/(4*w - chi_square)))/(w*r)*180/pi);
end
if r > 0.9
    d_unit_circle = ceil(acos(sqrt(w^2 - (w*r)^2)*exp(chi_square/w))/(w*r));
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% V-Test %%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for phi = 1:360
    V(phi) = w*r*cos(Theta_Ave - phi*pi/180);
    u(phi) = V(phi)*sqrt(2/w);

```

```

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Rao's Spacing Test %%%%%%%%%%%%%%
length_weighted_avg_angle = length_weighted_avg_angle*180/pi;
Theta_Ave = Theta_Ave*180/pi;
Protrusion_angle = Protrusion_angle*180/pi;
save('Angle of Protrusions.txt','Protrusion_angle','-ascii','-double');

lambda = 360/w; %degrees hereon
sorted_angles = sort(Protrusion_angle);
T(1) = sorted_angles(2) - sorted_angles(1);
for G = 2:(w - 1)
    T(G) = sorted_angles(G + 1) - sorted_angles(G);
end
T = T';
U_argument = abs(T - lambda);
U = 1/2*sum(U_argument);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Rayleigh's Test %%%%%%%%%%%%%%
z = w*r^2;
P = exp(sqrt(1 + 4*w + 4*(w^2 - (w*r)^2)) - (1 + 2*w));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Square displacement %%%%%%%%%%%%%%

MSD_from-Origin = (c_cell_center_x(:,1) - c_cell_center_x(1,1)).^2 + (c_cell_center_y(:,1) -
c_cell_center_y(1,1)).^2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
protrusion_length_over_time = zeros(i,1);
vector_magnitude = sqrt(Protrusion_lengths_from_membrane_x.^2 + Protrusion_lengths_from_membrane_y.^2);

for q = 1:i
    for m = 1:w
        if Protrusions_slice(m) == q
            %vector_magnitude(m,1)
            %protrusion_length_over_time(q,1)
            protrusion_length_over_time(q,1) = protrusion_length_over_time(q,1) + vector_magnitude(m,1);
        end
    end
    instantaneous_average_protrusion_length(q,1) = protrusion_length_over_time(q,1)/Protrusion_Histogram(q);
end

for count = 2:i
    MSD(count,1) = (c_cell_center_x(count,1) - c_cell_center_x(count - 1,1)).^2 + (c_cell_center_y(count,1) -
c_cell_center_y(count - 1,1)).^2;
end

protrusion_frequency = w/time_in_hours;

summary = [Theta_Ave length_weighted_avg_angle r_mean r_max w P U F r protrusion_frequency cell_radius X
Y]';
V_test = V';
u = u';
coordinates = [Protrusions_Unit_Circle_x Protrusions_Unit_Circle_y Protrusion_lengths_from_membrane_x
Protrusion_lengths_from_membrane_y];

```

```

for q = 1:i
    for m = 1:w
        if Protrusions_slice(m) == q
            Protrusion_vectors_from_membrane_x(q,1) = Protrusion_vectors_from_membrane_x(q,1) +
Protrusion_lengths_from_membrane_x(m,1);
            Protrusion_vectors_from_membrane_y(q,1) = Protrusion_vectors_from_membrane_y(q,1) +
Protrusion_lengths_from_membrane_y(m,1);
            Protrusion_unit_vectors_x(q,1) = Protrusion_unit_vectors_x(q,1) + Protrusions_Unit_Circle_x(m,1);
            Protrusion_unit_vectors_y(q,1) = Protrusion_unit_vectors_y(q,1) + Protrusions_Unit_Circle_y(m,1);
        end
    end

    if Protrusion_Histogram(q) > 0
        Instantaneous_Average_Vector_x(q,1) =
Protrusion_vectors_from_membrane_x(q,1)/Protrusion_Histogram(q);
        Instantaneous_Average_Vector_y(q,1) =
Protrusion_vectors_from_membrane_y(q,1)/Protrusion_Histogram(q);
        Instantaneous_Average_UnitVector_x(q,1) = Protrusion_unit_vectors_x(q,1)/Protrusion_Histogram(q);
        Instantaneous_Average_UnitVector_y(q,1) = Protrusion_unit_vectors_y(q,1)/Protrusion_Histogram(q);

        Instantaneous_Average_Angle(q,1) =
atan(Instantaneous_Average_Vector_y(q,1)/Instantaneous_Average_Vector_x(q,1));
        Instantaneous_Average_Unit_Angle(q,1) =
atan(Instantaneous_Average_UnitVector_y(q,1)/Instantaneous_Average_UnitVector_x(q,1));

        %Instantaneous_summed_angle(q,1) =
atan(Protrusion_vectors_from_membrane_y(q,1)/Protrusion_vectors_from_membrane_x(q,1));

        if Instantaneous_Average_Vector_x(q,1) < 0
            Instantaneous_Average_Angle(q,1) = Instantaneous_Average_Angle(q,1) + pi;
        end
        if Instantaneous_Average_Vector_x(q,1) > 0 && Instantaneous_Average_Vector_y(q,1) < 0
            Instantaneous_Average_Angle(q,1) = Instantaneous_Average_Angle(q,1) + 2*pi;
        end
        if Instantaneous_Average_Vector_x(q,1) == 0 && Instantaneous_Average_Vector_y(q,1) < 0
            Instantaneous_Average_Angle(q,1) = 3*pi/2;
        end
        if Instantaneous_Average_Vector_x(q,1) == 0 && Instantaneous_Average_Vector_y(q,1) > 0
            Instantaneous_Average_Angle(q,1) = pi/2;
        end

        if Instantaneous_Average_UnitVector_x(q,1) < 0
            Instantaneous_Average_Unit_Angle(q,1) = Instantaneous_Average_Unit_Angle(q,1) + pi;
        end
        if Instantaneous_Average_UnitVector_x(q,1) > 0 && Instantaneous_Average_UnitVector_y(q,1) < 0
            Instantaneous_Average_Unit_Angle(q,1) = Instantaneous_Average_Unit_Angle(q,1) + 2*pi;
        end
        if Instantaneous_Average_UnitVector_x(q,1) == 0 && Instantaneous_Average_UnitVector_y(q,1) < 0
            Instantaneous_Average_Unit_Angle(q,1) = 3*pi/2;
        end
        if Instantaneous_Average_UnitVector_x(q,1) == 0 && Instantaneous_Average_UnitVector_y(q,1) > 0
            Instantaneous_Average_Unit_Angle(q,1) = pi/2;
        end
    end
end

```



```

end
end

%[rows, cols] = find(Protrusion_Histogram == 0);
%Instantaneous_Average_Angle(rows) = [];
%Instantaneous_Average_Unit_Angle(rows) = [];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Time Correlation Function for Length Weighted Protrusions %%%%%%%%%%
Instantaneous_Average_Angle = Instantaneous_Average_Angle';
[lags_length_weighted, c_length_weighted] =
xcorr(Instantaneous_Average_Angle,Instantaneous_Average_Angle);
%lags_length_weighted = lags_length_weighted';
%[rows_length_weighted, columns_length_weighted] = find(lags_length_weighted == 0);
%lags_length_weighted(1:rows_length_weighted - 1) = [];
%c_length_weighted(1:rows_length_weighted - 1) = [];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Time Correlation Function for UnitCircle Protrusions %%%%%%%%%%
Instantaneous_Average_Unit_Angle = Instantaneous_Average_Unit_Angle';
[lags_UnitCircle, c_UnitCircle] = cxcorr(Instantaneous_Average_Unit_Angle,Instantaneous_Average_Unit_Angle);
%lags_UnitCircle = lags_UnitCircle';
%[rows_UnitCircle, columns_UnitCircle] = find(lags_UnitCircle == 0);
%lags_UnitCircle(1:rows_UnitCircle - 1) = [];
%c_UnitCircle(1:rows_UnitCircle - 1) = [];
figure()
plot(lags_length_weighted, c_length_weighted,'-r',lags_UnitCircle, c_UnitCircle,'-b',lags_length_weighted,
c_length_weighted,'or',lags_UnitCircle, c_UnitCircle,'ob')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Fourier Transform of Time Correlation Functions %%%%%%%%%%
fs = (1/60*interval);
NFFT = 2^nextpow2(length(Instantaneous_Average_Angle));
F_length_weighted = fft(c_length_weighted, NFFT);
x = (0:NFFT-1)*(fs/NFFT);
F_length_weighted = F_length_weighted.*conj(F_length_weighted)/NFFT; %
2*abs(F_length_weighted(1:NFFT/2+1));

F_UnitCircle = fft(c_UnitCircle, NFFT);
F_UnitCircle = F_UnitCircle.*conj(F_UnitCircle)/NFFT; % 2*abs(F_UnitCircle(1:NFFT/2+1));

figure()
plot(x,F_length_weighted,'or',x,F_UnitCircle,'ob',x,F_length_weighted,'-r',x,F_UnitCircle,'-b')
xlabel('Frequency (Hz)')
ylabel('Power')

Instantaneous_Average_Angle = Instantaneous_Average_Angle';
Instantaneous_Average_Unit_Angle = Instantaneous_Average_Unit_Angle';
lags_UnitCircle = lags_UnitCircle';
lags_length_weighted = lags_length_weighted';
c_length_weighted = c_length_weighted';
c_UnitCircle = c_UnitCircle';
F_UnitCircle = F_UnitCircle';
F_length_weighted = F_length_weighted';

save('Time_in_Hours.txt', 'time','-ascii', '-double');

```

```

save('Square_Displacement_from_Origin.txt','MSD_from_Origin','-ascii','-double');
save('Binned_protrusions_360.txt','Binned_protrusions_360','-ascii','-double');
save('SummaryMatrix.txt','summary','-ascii','-double');
save('Unit_Circle_x.txt','Protrusions_Unit_Circle_x','-ascii','-double');
save('Unit_Circle_y.txt','Protrusions_Unit_Circle_y','-ascii','-double');
save('x-Coordinate_Protrusion_Lengths_from_Cell_Membrane.txt','Protrusion_lengths_from_membrane_x','-ascii','-double');
save('y-Coordinate_Protrusion_Lengths_from_Cell_Membrane.txt','Protrusion_lengths_from_membrane_y','-ascii','-double');
%save('V-Test.txt','V_test','-ascii','-double');
save('Critical_Value_for_V-Test (u).txt','u','-ascii','-double');
save('Protrusion_Histogram_with_Time.txt','Protrusion_Histogram','-ascii','-double');
save('x-Fourier.txt','x','-ascii','-double');
save('Fourier_LW_TimeCorrelation.txt','F_length_weighted','-ascii','-double');
save('Fourier_UnitCircle_TimeCorrelation.txt','F_UnitCircle','-ascii','-double');
save('lags_LW','lags_length_weighted','-ascii','-double');
save('lags_UnitCircle','lags_UnitCircle','-ascii','-double');
save('Correlation_Sequence_LW.txt','c_length_weighted','-ascii','-double');
save('Correlation_Sequence_UnitCircle.txt','c_UnitCircle','-ascii','-double');
save('Instantaneous_Average_Angle.txt','Instantaneous_Average_Angle','-ascii','-double');
save('Instantaneous_Average_Unit_Angle.txt','Instantaneous_Average_Unit_Angle','-ascii','-double');
save('Protrusion_Length_Over_Time.txt','instantaneous_average_protrusion_length','-ascii','-double');
save('Square_Displacement.txt','MSD','-ascii','-double');

```

```

%save('Binned_protrusions_180.txt','Binned_protrusions_180','-ascii','-double');
%save('Maximum_Protrusion_Length.txt','r_max','-ascii','-double');
%save('Average_Protrusion_Length.txt','r_mean','-ascii','-double');
%save('Critical_Value_for_V-Test.txt','u','-ascii','-double');
%save('V-Test_V.txt','V','-ascii','-double');
%save('Raos_Spacing_Test.txt','U','-ascii','-double');
%save('Number_of_Protrusions.txt','w','-ascii','-double');
%save('Global_Protrusion_Clustering_Index.txt','r','-ascii','-double');
%save('Global_Average_Protrusion_Angle.txt','Theta_Ave','-ascii','-double');
%save('Global_Average_Protrusion_Length_Weigthed_Angle.txt','length_weighted_avg_angle','-ascii','-double');
%save('Cell_Radius.txt','cell_radius','-ascii','-double');

```

```

%%%%%%%%%% EARLY TIME ANALYSIS
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%
%%%%%%%%%%

```

```

rows_early = find(Protrusions_slice <= 60); %round(length(Uncorrected_Protrusions_x)/2));
x_P_double_prime_early = Protrusion_lengths_from_membrane_x(rows_early);
y_P_double_prime_early = Protrusion_lengths_from_membrane_y(rows_early);
w_early = length(rows_early)
protrusion_frequency = w_early/10;

```

```

%%%%%%%% Calculating angle of protrusion in the appropriate coordinate
%%%%%%%% system and correcting for quadrant location

Protrusion_angle_early = atan(y_P_double_prime_early./x_P_double_prime_early);
Protrusions_Unit_Circle_x_early = cos(Protrusion_angle_early);
Protrusions_Unit_Circle_y_early = sin(Protrusion_angle_early);

for m = 1:w_early

    if x_P_double_prime_early(m,1) < 0
        Protrusion_angle_early(m,1) = Protrusion_angle_early(m,1) + pi;
        Protrusions_Unit_Circle_x_early(m,1) = cos(Protrusion_angle_early(m,1));
        Protrusions_Unit_Circle_y_early(m,1) = sin(Protrusion_angle_early(m,1));
    end
    if x_P_double_prime_early(m,1) > 0 && y_P_double_prime_early(m,1) < 0
        Protrusion_angle_early(m,1) = Protrusion_angle_early(m,1) + 2*pi;
        Protrusions_Unit_Circle_x_early(m,1) = cos(Protrusion_angle_early(m,1));
        Protrusions_Unit_Circle_y_early(m,1) = sin(Protrusion_angle_early(m,1));
    end
    if x_P_double_prime_early(m,1) == 0 && y_P_double_prime_early(m,1) < 0
        Protrusion_angle_early(m,1) = 3*pi/2;
        Protrusions_Unit_Circle_x_early(m,1) = cos(Protrusion_angle_early(m,1));
        Protrusions_Unit_Circle_y_early(m,1) = sin(Protrusion_angle_early(m,1));
    end
    if x_P_double_prime_early(m,1) == 0 && y_P_double_prime_early(m,1) > 0
        Protrusion_angle_early(m,1) = pi/2;
        Protrusions_Unit_Circle_x_early(m,1) = cos(Protrusion_angle_early(m,1));
        Protrusions_Unit_Circle_y_early(m,1) = sin(Protrusion_angle_early(m,1));
    end
end

circ_plot(Protrusion_angle_early);
title('Protrusion Angles Early');
save('Protrusion_angles_early.txt','Protrusion_angle_early','-ascii','-double');

Protrusion_lengths_from_membrane_x_early = x_P_double_prime_early;
Protrusion_lengths_from_membrane_y_early = y_P_double_prime_early;

%%%%%%%%

%%%%%%%% Resultant Lenght for Set of Angles on Unit Circle %%%%%%%%%

X_early = (1/w_early)*sum(Protrusions_Unit_Circle_x_early);
Y_early = (1/w_early)*sum(Protrusions_Unit_Circle_y_early);

r_early = sqrt(X_early^2 + Y_early^2);

%%%%%%%%

%%%%%%%% Resultant Length for Set of Angles Length-based %%%%%%%%%

X_length_early = (1/w_early)*sum(Protrusion_lengths_from_membrane_x_early);
Y_length_early = (1/w_early)*sum(Protrusion_lengths_from_membrane_y_early);

```

```

r_length_early = sqrt(X_length_early^2 + Y_length_early^2);

r1_early = sqrt(Protrusion_lengths_from_membrane_x_early.^2 +
Protrusion_lengths_from_membrane_y_early.^2);
r_max_early = max(r1_early);
r_mean_early = mean(r1_early);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Moore modified Rayleigh
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
vectors_early = [r1_early Protrusion_angle_early];
sorted_lengths_from_membrane_early = sortrows(vectors_early);
n_early = (1:w_early)';
X_Moore_early = sum(n_early.*cos(vectors_early(:,2)));
Y_Moore_early = sum(n_early.*sin(vectors_early(:,2)));
R_Moore_early = sqrt(X_Moore_early^2 + Y_Moore_early^2);
R_critical_Moore_early = R_Moore_early/(w_early^(3/2));
Double_Tail_R_critical_early = 2*R_critical_Moore_early;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Parametric One-Sample Second-Order Analysis %%%%%%%%%%
normalized_r1_early = r1_early/max(r1_early);
normalized_Protrusion_lengths_from_membrane_x_early = normalized_r1_early.*cos(Protrusion_angle_early);
normalized_Protrusion_lengths_from_membrane_y_early = normalized_r1_early.*sin(Protrusion_angle_early);
normalized_X_length_early = (1/w_early)*sum(normalized_Protrusion_lengths_from_membrane_x_early);
normalized_Y_length_early = (1/w_early)*sum(normalized_Protrusion_lengths_from_membrane_y_early);
sum_x_square_early = sum(normalized_Protrusion_lengths_from_membrane_x_early.^2) -
(1/w_early)*(sum(normalized_Protrusion_lengths_from_membrane_x_early))^2;
sum_y_square_early = sum(normalized_Protrusion_lengths_from_membrane_y_early.^2) -
(1/w_early)*(sum(normalized_Protrusion_lengths_from_membrane_y_early))^2;
sum_xy_early =
sum(normalized_Protrusion_lengths_from_membrane_x_early.*normalized_Protrusion_lengths_from_membrane_y_early) -
(1/w_early)*(sum(normalized_Protrusion_lengths_from_membrane_x_early)*(sum(normalized_Protrusion_lengths_from_membrane_y_early)));
F_early = (w_early/2)*(w_early - 2)*(normalized_X_length_early^2*sum_y_square_early -
2*normalized_X_length_early*normalized_Y_length_early*sum_xy_early +
normalized_Y_length_early^2*sum_x_square_early)/(sum_x_square_early*sum_y_square_early -
sum_xy_early^2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Binning %%%%%%%%%%
Binned_protrusions_30_early = zeros(12,1);

for m = 1:w_early

    if 0 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= pi/6
        Binned_protrusions_30_early(1) = Binned_protrusions_30_early(1) + 1;
    end

    if pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= pi/3
        Binned_protrusions_30_early(2) = Binned_protrusions_30_early(2) + 1;
    end

    if pi/3 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= pi/2
        Binned_protrusions_30_early(3) = Binned_protrusions_30_early(3) + 1;
    end

```

```

    if pi/2 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 2*pi/3
        Binned_protrusions_30_early(4) = Binned_protrusions_30_early(4) + 1;
    end

    if 2*pi/3 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 5*pi/6
        Binned_protrusions_30_early(5) = Binned_protrusions_30_early(5) + 1;
    end

    if 5*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= pi
        Binned_protrusions_30_early(6) = Binned_protrusions_30_early(6) + 1;
    end

    if pi < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 7*pi/6
        Binned_protrusions_30_early(7) = Binned_protrusions_30_early(7) + 1;
    end

    if 7*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 8*pi/6
        Binned_protrusions_30_early(8) = Binned_protrusions_30_early(8) + 1;
    end

    if 8*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 9*pi/6
        Binned_protrusions_30_early(9) = Binned_protrusions_30_early(9) + 1;
    end

    if 9*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 10*pi/6
        Binned_protrusions_30_early(10) = Binned_protrusions_30_early(10) + 1;
    end

    if 10*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 11*pi/6
        Binned_protrusions_30_early(11) = Binned_protrusions_30_early(11) + 1;
    end

    if 11*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 2*pi
        Binned_protrusions_30_early(12) = Binned_protrusions_30_early(12) + 1;
    end
end

%%%%%% Average Angle Calculations (Length Weighted and Unit Circle) %%%%%%

length_weighted_avg_angle_early = atan(Y_length_early/X_length_early);

if X_length_early < 0
    length_weighted_avg_angle_early = length_weighted_avg_angle_early + pi;
end
if X_length_early > 0 && Y_length_early < 0
    length_weighted_avg_angle_early = length_weighted_avg_angle_early + 2*pi;
end
if X_length_early == 0 && Y_length_early < 0
    length_weighted_avg_angle_early = 3*pi/2;
end
if X_length_early == 0 && Y_length_early > 0
    length_weighted_avg_angle_early = pi/2;
end

```

```

Theta_Ave_early = atan(Y_early/X_early);

if X_early < 0
    Theta_Ave_early = Theta_Ave_early + pi;
end
if X_early > 0 && Y_early < 0
    Theta_Ave_early = Theta_Ave_early + 2*pi;
end
if X_early == 0 && Y_early < 0
    Theta_Ave_early = 3*pi/2;
end
if X_early == 0 && Y_early > 0
    Theta_Ave_early = pi/2;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Confidence Interval for Average Angles %%%%%%%%%
chi_square = 3.841; %95 percent confidence_interval
if r_early <= 0.9 && r_early > sqrt(chi_square/(2*w_early))
    d_unit_circle_early = ceil(acos(sqrt((2*w_early*(2*(w_early*r_early)^2 - w_early*chi_square)/(4*w_early -
chi_square)))/w_early*r_early)*180/pi);
end
if r_early > 0.9
    d_unit_circle_early = ceil(acos(sqrt(w_early^2 - (w_early^2 -
(w_early*r_early)^2)*exp(chi_square/w_early))/(w_early*r_early)));
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% V-Test %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for phi = 1:360
    V_early(phi) = w*r_early*cos(Theta_Ave_early - phi*pi/180);
    u_early(phi) = V_early(phi)*sqrt(2/w_early);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Rao's Spacing Test %%%%%%%%%
length_weighted_avg_angle_early = length_weighted_avg_angle_early*180/pi;
Theta_Ave_early = Theta_Ave_early*180/pi;
Protrusion_angle_early = Protrusion_angle_early*180/pi;

lambda_early = 360/w_early; %degrees hereon
sorted_angles_early = sort(Protrusion_angle_early);
T_early(1) = sorted_angles_early(2) - sorted_angles_early(1);
for a = 2:(w_early - 1)
    T_early(a) = sorted_angles_early(a + 1) - sorted_angles_early(a);
end
T_early = T_early';
U_argument_early = abs(T_early - lambda_early);
U_early = 1/2*sum(U_argument_early);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Rayleigh's Test %%%%%%%%%
z_early = w_early*r_early^2;
P_early = exp(sqrt(1 + 4*w_early + 4*(w_early^2 - (w_early*r_early)^2)) - (1 + 2*w_early));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```



```

        Protrusion_angle_late(m,1) = 3*pi/2;
        Protrusions_Unit_Circle_x_late(m,1) = cos(Protrusion_angle_late(m,1));
        Protrusions_Unit_Circle_y_late(m,1) = sin(Protrusion_angle_late(m,1));
    end
    if x_P_double_prime_late(m,1) == 0 && y_P_double_prime_late(m,1) > 0
        Protrusion_angle_late(m,1) = pi/2;
        Protrusions_Unit_Circle_x_late(m,1) = cos(Protrusion_angle_late(m,1));
        Protrusions_Unit_Circle_y_late(m,1) = sin(Protrusion_angle_late(m,1));
    end
end

circ_plot(Protrusion_angle_late);
title('Protrusion Angles Late');
save('Protrusion_angles_late.txt','Protrusion_angle_late','-ascii','-double');

Protrusion_lengths_from_membrane_x_late = x_P_double_prime_late; % -
cell_radius*cos(Protrusion_delta_Theta(m,1));
Protrusion_lengths_from_membrane_y_late = y_P_double_prime_late; % -
cell_radius*sin(Protrusion_delta_Theta(m,1));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Resultant Lenght for Set of Angles on Unit Circle %%%%%%%%%%

X_late = (1/w_late)*sum(Protrusions_Unit_Circle_x_late);
Y_late = (1/w_late)*sum(Protrusions_Unit_Circle_y_late);

r_late = sqrt(X_late^2 + Y_late^2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Resultant Length for Set of Angles Length-based %%%%%%%%%%

X_length_late = (1/w_late)*sum(Protrusion_lengths_from_membrane_x_late);
Y_length_late = (1/w_late)*sum(Protrusion_lengths_from_membrane_y_late);

r_length_late = sqrt(X_length_late^2 + Y_length_late^2);

r1_late = sqrt(Protrusion_lengths_from_membrane_x_late.^2 + Protrusion_lengths_from_membrane_y_late.^2);
r_max_late = max(r1_late);
r_mean_late = mean(r1_late);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Moore modified Rayleigh
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
vectors_late = [r1_late Protrusion_angle_late];
sorted_lengths_from_membrane_late = sortrows(vectors_late);
n_late = (1:w_late)';
X_Moore_late = sum(n_late.*cos(vectors_late(:,2)));
Y_Moore_late = sum(n_late.*sin(vectors_late(:,2)));
R_Moore_late = sqrt(X_Moore_late^2 + Y_Moore_late^2);
R_critical_Moore_late = R_Moore_late/(w_late^(3/2));
Double_Tail_R_critical_late = 2*R_critical_Moore_late;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Parametric One-Sample Second-Order Analysis %%%%%%%%%%
normalized_r1_late = r1_late/max(r1_late);

```



```

normalized_Protrusion_lengths_from_membrane_x_late = normalized_r1_late.*cos(Protrusion_angle_late);
normalized_Protrusion_lengths_from_membrane_y_late = normalized_r1_late.*sin(Protrusion_angle_late);
normalized_X_length_late = (1/w_late)*sum(normalized_Protrusion_lengths_from_membrane_x_late);
normalized_Y_length_late = (1/w_late)*sum(normalized_Protrusion_lengths_from_membrane_y_late);
sum_x_square_late = sum(normalized_Protrusion_lengths_from_membrane_x_late.^2) -
(1/w_late)*(sum(normalized_Protrusion_lengths_from_membrane_x_late))^2;
sum_y_square_late = sum(normalized_Protrusion_lengths_from_membrane_y_late.^2) -
(1/w_late)*(sum(normalized_Protrusion_lengths_from_membrane_y_late))^2;
sum_xy_late =
sum(normalized_Protrusion_lengths_from_membrane_x_late.*normalized_Protrusion_lengths_from_membrane_y_
late) -
(1/w_late)*(sum(normalized_Protrusion_lengths_from_membrane_x_late)*(sum(normalized_Protrusion_lengths_fr
om_membrane_y_late)));
F_late = (w_late/2)*(w_late - 2)*(normalized_X_length_late^2*sum_y_square_late -
2*normalized_X_length_late*normalized_Y_length_late*sum_xy_late +
normalized_Y_length_late^2*sum_x_square_late)/(sum_x_square_late*sum_y_square_late - sum_xy_late^2);

```

%%%%%%%%%%%%%% Binning %%%%%%%%%%%%%%%

```

Binned_protrusions_30_late = zeros(12,1);

```

```

for m = 1:w_late

```

```

    if 0 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= pi/6
        Binned_protrusions_30_late(1) = Binned_protrusions_30_late(1) + 1;
    end

```

```

    if pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= pi/3
        Binned_protrusions_30_late(2) = Binned_protrusions_30_late(2) + 1;
    end

```

```

    if pi/3 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= pi/2
        Binned_protrusions_30_late(3) = Binned_protrusions_30_late(3) + 1;
    end

```

```

    if pi/2 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 2*pi/3
        Binned_protrusions_30_late(4) = Binned_protrusions_30_late(4) + 1;
    end

```

```

    if 2*pi/3 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 5*pi/6
        Binned_protrusions_30_late(5) = Binned_protrusions_30_late(5) + 1;
    end

```

```

    if 5*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= pi
        Binned_protrusions_30_late(6) = Binned_protrusions_30_late(6) + 1;
    end

```

```

    if pi < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 7*pi/6
        Binned_protrusions_30_late(7) = Binned_protrusions_30_late(7) + 1;
    end

```

```

    if 7*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 8*pi/6
        Binned_protrusions_30_late(8) = Binned_protrusions_30_late(8) + 1;
    end

```

```

    if 8*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 9*pi/6
        Binned_protrusions_30_late(9) = Binned_protrusions_30_late(9) + 1;
    end

    if 9*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 10*pi/6
        Binned_protrusions_30_late(10) = Binned_protrusions_30_late(10) + 1;
    end

    if 10*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 11*pi/6
        Binned_protrusions_30_late(11) = Binned_protrusions_30_late(11) + 1;
    end

    if 11*pi/6 < Protrusion_angle(m,1) && Protrusion_angle(m,1) <= 2*pi
        Binned_protrusions_30_late(12) = Binned_protrusions_30_late(12) + 1;
    end
end

%%%%%%%% Average Angle Calculations (Length Weighted and Unit Circle) %%%%%%%%%

length_weighted_avg_angle_late = atan(Y_length_late/X_length_late);

if X_length_late < 0
    length_weighted_avg_angle_late = length_weighted_avg_angle_late + pi;
end
if X_length_late > 0 && Y_length_late < 0
    length_weighted_avg_angle_late = length_weighted_avg_angle_late + 2*pi;
end
if X_length_late == 0 && Y_length_late < 0
    length_weighted_avg_angle_late = 3*pi/2;
end
if X_length_late == 0 && Y_length_late > 0
    length_weighted_avg_angle_late = pi/2;
end

Theta_Ave_late = atan(Y_late/X_late);

if X_late < 0
    Theta_Ave_late = Theta_Ave_late + pi;
end
if X_late > 0 && Y_late < 0
    Theta_Ave_late = Theta_Ave_late + 2*pi;
end
if X_late == 0 && Y_late < 0
    Theta_Ave_late = 3*pi/2;
end
if X_late == 0 && Y_late > 0
    Theta_Ave_late = pi/2;
end

%%%%%%%% Confidence Interval for Average Angles %%%%%%%%%
chi_square = 3.841; %95 percent confidence_interval
if r_late <= 0.9 && r_late > sqrt(chi_square/(2*w_late))
    d_unit_circle_late = ceil(acos(sqrt((2*w_late*(2*(w_late*r_late)^2 - w_late*chi_square)/(4*w_late -
chi_square)))/w_late*r_late)*180/pi);

```

```

end
if r_late > 0.9
    d_unit_circle_late = ceil(acos(sqrt(w_late^2 - (w_late^2 -
(w_late*r_late)^2)*exp(chi_square/w_late))/(w_late*r_late)));
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% V-Test
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for phi = 1:360
    V_late(phi) = w_late*r_late*cos(Theta_Ave_late - phi*pi/180);
    u_late(phi) = V_late(phi)*sqrt(2/w_late);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Rao's Spacing Test %%%%%%%%%%
length_weighted_avg_angle_late = length_weighted_avg_angle_late*180/pi;
Theta_Ave_late = Theta_Ave_late*180/pi;
Protrusion_angle_late = Protrusion_angle_late*180/pi;

lambda_late = 360/w_late;    %degrees hereon
sorted_angles_late = sort(Protrusion_angle_late);
T_late(1) = sorted_angles_late(2) - sorted_angles_late(1);
for a = 2:(w_late - 1)
    T_late(a) = sorted_angles_late(a + 1) - sorted_angles_late(a);
end
T_late = T_late';
U_argument_late = abs(T_late - lambda_late);
U_late = 1/2*sum(U_argument_late);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Rayleigh's Test %%%%%%%%%%
z_late = w_late*r_late^2;
P_late = exp(sqrt(1 + 4*w_late + 4*(w_late^2 - (w_late*r_late)^2)) - (1 + 2*w_late));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

summary_late = [Theta_Ave_late length_weighted_avg_angle_late r_mean_late r_max_late w_late P_late U_late
F_late r_late protrusion_frequency cell_radius X_late Y_late]';
V_test_late = V_late';
u_late = u_late';
coordinates = [Protrusions_Unit_Circle_x Protrusions_Unit_Circle_y Protrusion_lengths_from_membrane_x
Protrusion_lengths_from_membrane_y];

save('SummaryMatrix_late.txt','summary_late','-ascii','-double');
save('Unit_Circle_x_late.txt','Protrusions_Unit_Circle_x_late','-ascii','-double');
save('Unit_Circle_y_late.txt','Protrusions_Unit_Circle_y_late','-ascii','-double');
save('x-
Coordinate_Protrusion_Lenghts_from_Cell_Membrane_late.txt','Protrusion_lengths_from_membrane_x_late',
'-ascii','-double');
save('y-
Coordinate_Protrusion_Lenghts_from_Cell_Membrane_late.txt','Protrusion_lengths_from_membrane_y_late',
'-ascii','-double');
save('Critical_Value_for_V-Test(u)_late.txt','u_late','-ascii','-double');
save('Protrusion_Histogram_Bin30_late.txt','Binned_protrusions_30_late','-ascii','-double');

save('corrected_cell_center_x.txt','c_cell_center_x','-ascii','-double');
save('corrected_cell_center_y.txt','c_cell_center_y','-ascii','-double');

```

```
save('Confidence_Interval_for_AverageAngle_95%Confidence.txt','d_unit_circle','-ascii','-double');
```

D-5 Anastomosis image analysis

```
clear
clc

img_path0 = strcat('G:\Anthony EHD Research Files\PIVLT\Vertical Format - Invasions and
vasculogenesis\((ADS3_5-6-2012) 7days vertical anastomosis\Monolayer Only\R-C_3');
img_path1 = strcat(img_path0,'Intensities - corrected');
img_path2 = strcat(img_path1,'\*.tif');

file = dir(img_path2);
file_num = length(file);
file_name = {file.name};
file_date = {file.date};

cd(img_path1);

for i = 1:file_num
    RGB = imread(file_name{i});

    R = RGB(:, :, 1); %selects only the red channel
    G = RGB(:, :, 2); %selects only the green channel

    Intensity_count_R(i) = sum(R(:)); %stores the intensity values of every slice
    Intensity_count_G(i) = sum(G(:));

end

stack_intensity_fraction_per_slice_R = Intensity_count_R/sum(Intensity_count_R); %normalizes the intensities
against the total stack
stack_intensity_fraction_per_slice_G = Intensity_count_G/sum(Intensity_count_G);

stack_intensity_fraction_per_slice_G = stack_intensity_fraction_per_slice_G';
stack_intensity_fraction_per_slice_R = stack_intensity_fraction_per_slice_R';

save('Red fraction per slice.txt','stack_intensity_fraction_per_slice_R','-ascii','-double');
save('Green fraction per slice.txt','stack_intensity_fraction_per_slice_G','-ascii','-double');

beep
```